Installation and Configuration Guide

Sun™ ONE Identity Synchronization for Windows

Version 1.0

816-6393-10
December 2003

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 U.S.A.

Copyright 2003 Sun Microsystems, Inc. All rights reserved.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, Sun ONE, iPlanet, and all Sun, Java, and Sun ONE based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

Netscape is a trademark or registered trademark of Netscape Communications Corporation in the United States and other countries.

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of the product or this document may be reproduced in any form by any means without prior written authorization of Sun Microsystems, Inc. and its licensors, if any.

THIS DOCUMENTATION IS PROVIDED “AS IS” AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2003 Sun Microsystems, Inc. Tous droits réservés.

Sun, Sun Microsystems, le logo Sun, Java, Solaris, Sun ONE, et iPlanet sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et d’autre pays.

UNIX est une marque enregistree aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company Ltd.

Netscape est une marque de Netscape Communications Corporation aux Etats-Unis et dans d'autres pays.

Le produit décrit dans ce document est distribué selon des conditions de licence qui en restreignent l'utilisation, la copie, la distribution et la décompilation. Aucune partie de ce produit ni de ce document ne peut être reproduite sous quelque forme ou par quelque moyen que ce soit sans l’autorisation écrite préalable de Sun Microsystems, Inc. et, le cas échéant, de ses bailleurs de licence.

CETTE DOCUMENTATION EST FOURNIE “EN L'ÉTAT”, ET TOUTES CONDITIONS EXPRESSES OU IMPLICITES, TOUTES REPRÉSENTATIONS ET TOUTES GARANTIES, Y COMPRIS TOUTE GARANTIE IMPLICITE D'APTITUDE À LA VENTE, OU À UN BUT PARTICULIER OU DE NON CONTREFAÇON SONT EXCLUES, EXCEPTÉ DANS LA MESURE OÙ DE TELLES EXCLUSIONS SERAIENT CONTRAIRES À LA LOI.

Contents

About This Guide

Sun[tm] Identity Synchronization for Windows allows password and other specified user attributes to flow between Sun ONE Directory Server and other systems. Sun ONE Identity Synchronization for Windows software is part of the Sun Open Net Environment (Sun ONE), Sun's standards-based software vision, architecture, platform, and expertise for building and deploying Services On Demand.

Purpose of This Guide

This guide demonstrates how to install Identity Synchronization for Windows for use in a production environment. Preparing Identity Synchronization for Windows for performance in a production environment also involves considerable configuration.

For the latest information about new features and enhancements in this release of Identity Synchronization for Windows, please see the online release notes at:

```
http://www.sun.com/
```

NOTE User interfaces depicted in this document are subject to change in future versions of the product.

Conventions Used in This Book

This section explains the typographic conventions used in this book.

`Monospaced font` **-** This typeface is used for literal text, such as the names of attributes and object classes when they appear in text. It is also used for URLs, filenames, and examples.

Italic font **-** This typeface is used for emphasis, for new terms, and for text that you must substitute for actual values, such as placeholders in path names.

The greater than symbol (>) is used as a separator for successive menu selections. For example, Object > New > User means that you should pull down the Object menu, drag the mouse down to highlight New, and drag the mouse across to the New submenu in which you must select User.

NOTE Notes, Cautions, and Tips highlight important conditions or limitations. Be sure to read this information before continuing.

This book uses the following format for paths and file names:

installDir`/slapd-`*serverID*`/...`

The actual path and server identifier will depend on your platform, your installation, and your configuration. The default path is platform-dependent:

Solaris 9 platform	`/var/ds5/slapd-`*serverID*`/...`
Other UNIX platforms	`/usr/sun/servers/slapd-`*serverID*`/...`
Windows platform	`C:\Program Files\Sun\MPS\slapd-`*serverID*`\...`

If you have installed Directory Server in a different location, you should adapt the path accordingly. *serverID* represents the server identifier you gave the server when you installed it. For example, if you gave the name phonebook to your Registry server, then the actual path would be:

Solaris 9 platform	`/var/ds5/slapd-phonebook/...`
Other UNIX platforms	`/usr/sun/servers/slapd-phonebook/...`
Windows platform	`C:\Program Files\Sun\MPS\slapd-phonebook\...`

Most paths and commands specified in this manual are in UNIX format. If you are using a Windows-based version of Directory Server, use equivalent paths and commands. All commands on Windows platforms have the same name with the `.exe` or `.bat` extension.

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

NOTE Sun is not responsible for the availability of third-party Web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Related Information

Other useful Sun ONE information can be found at the following Internet locations:

- Product documentation on line—`http://docs.sun.com`
- Product support and status—`http://www.sun.com/service/support/software/`
- Sun Enterprise Services for Solaris patches and support—`http://www.sun.com/service/`
- Developer information—`http://www.sun.com/developers/`
- Support and Training—`http://www.sun.com/supportraining/`
- Product data sheets—`http://www.sun.com/software/`

Accessibility Features

Based on the Java^{TM} Foundation Classes (JFC), the Identity Synchronization for Windows console provides support for the assistive software and technologies that make software accessible to users with disabilities. This section describes the accessibility features of the Identity Synchronization for Windows console, and the improvements that have been made to the document set to make it more accessible.

Console Accessibility Features

Most of the accessibility features described in the following section are provided automatically through the use of JFC/Swing! components.

Accessible Names And Descriptions

All objects have accessible names (succinct explanations of the object's purpose). These names can be used by assistive technologies to present the objects to the user. Accessible descriptions are more verbose explanations that provide additional information on objects, where this is necessary.

Customizable Fonts

The style and size of fonts in text panes, menus, labels, and information messages, can be customized.

Although color coding is used to convey information, it is not the only means of doing so.

Dynamic GUI Layout

The dynamic layout allows users to specify the size and position of Directory Server windows, or for this to be determined by the user's settings.

Keyboard Traversable Components

This accessibility feature caters for users who have difficulty using a mouse. Pressing the tab key moves the input focus from component to component and shift-tab moves the focus in the opposite direction. The arrow keys allow users to navigate trees without using the mouse.

The focus is programmatically exposed so that assistive software can track focus and focus changes.

Text Equivalents for Non-text Elements

When an image represents a program element, the information conveyed by the image is also available in the text.

Equivalent Command-line Interface

Most of the functionality of the console can be achieved at the command line.

Documentation Accessibility Features

The Identity Synchronization for Windows 1.0 document set is delivered in both PDF and HTML format. This section describes accessibility features in the HTML version of the documentation.

Text Equivalents for Non-text Elements

Alternative text labels are assigned to links or graphics. Where graphics provide detailed descriptions, text versions of these descriptions are provided either within the surrounding text, or in a separate file.

Tables That Can Be Interpreted By Assistive Technology

All tables now include descriptive headers. A brief description of the table contents is also provided in the surrounding text.

Part 1

Installation and Configuration

Chapter 1

Understanding the Product

To help you prepare for your Sun ONE Identity Synchronization for Windows installation, you should be familiar with the concepts contained in the following sections:

- System Components
- Command Line Interfaces
- System Components Distribution
- Deployment Example: A Two-Machine Configuration

Identity Synchronization for Windows provides bidirectional password synchronization between the Sun ONE Directory Server 5.2 and:

- Windows 2000 Active Directory
- Windows NT SAM Registry.

Users accessing applications that use these directories for login authentication need only remember a single password, and when applying periodic password updates, the user is only required to make the password update once. In addition, product features include:

- Synchronization of selected additional attributes in the user entry, whenever any selected attribute is modified in one directory environment the new values are immediately and automatically propagated to the other directory
- User account creation synchronization between the supported directories; whenever a user is created in one directory environment the new values are immediately and automatically propagated to the other directory

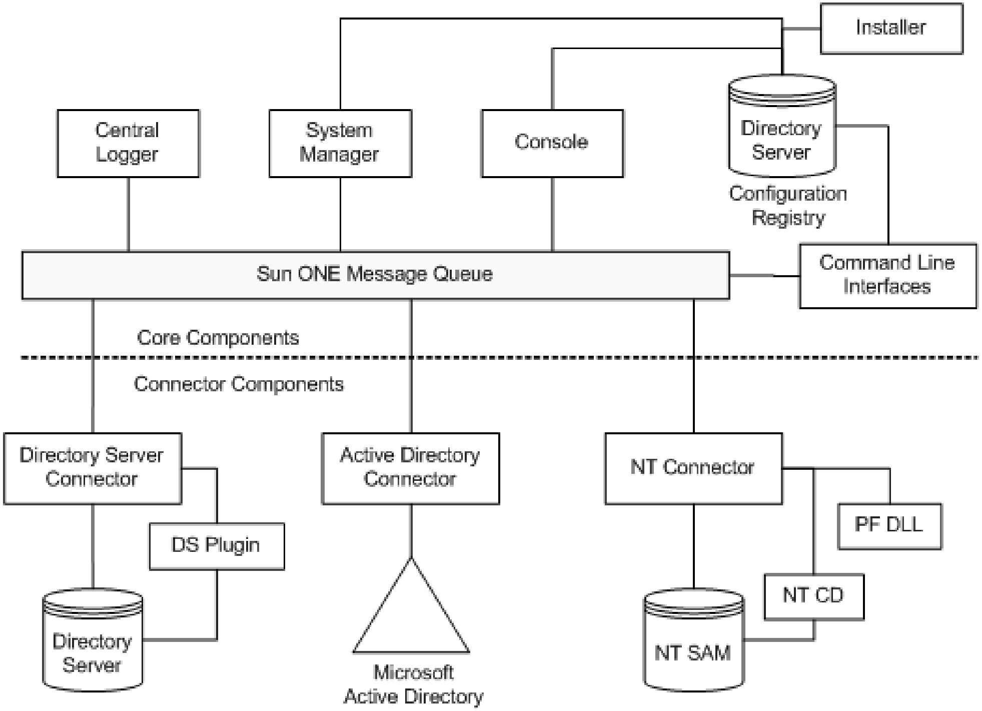
There is no requirement to modify the Windows directories, or to change the applications using the directories.

When using the product between Sun ONE Directory Server and Active Directory there is no requirement to install any components in the Windows operating environment. When synchronizing between Sun ONE Directory Server and Windows NT, the product's NT component must be run in the Windows NT environment.

System Components

Identity Synchronization for Windows is made out of a set of core components and any number of individual connectors and connector subcomponents that allow for the synchronization of password updates between Sun ONE and Windows directories (see Figure 1-1).

Figure 1-1 System Components



Core

Core includes the following components: console, system manager, central logger, Sun ONE Message Queue, and the product’s configuration registry, which is stored in a Sun ONE Directory Server Configuration Directory.

Product’s Configuration Registry

The product’s configuration registry is stored in a Directory Server configuration directory. The console, system manager and the install3r all read and write the product’s configuration data from the configuration registry. This registry contains:

- Configuration information for every directory, domain, connector and connector sub-component
- Default log settings
- Synchronization settings describing the direction of user creation and attribute modifications
- Attributes to be synchronized and attribute mappings between the two directory environments
- Synchronized user lists in each directory topology

Console

The console centralizes all configuration and administration of the product’s components, it allows the user to:

- Configure directories or domains being synchronized
- Select user entry attributes to be synchronized, in addition to passwords
- Specify list of users within each directory or domain topology that require synchronization
- Monitor system status
- Dynamically and selectively stop and start password synchronization for different directories or domains

Selected console actions can also be performed via the product’s command line interfaces

System Manager

- Leverages the product's backend networked facilities for dynamically delivering configuration updates to connectors
- Keeps status on each connector and connector subcomponents

Central Logger

Connectors might be installed so that they are widely distributed across remote geographical locations; therefore, it is of great administrative value to have all logging information be centralized allowing the Administrator to easily monitor synchronization activity, detect errors and evaluate the health of the entire system from a single network location.

The central logger logs allow a system administrator to,

- Verify that the system is running correctly
- Detect and resolve individual component and system-wide problems
- Audit individual and system-wide synchronization activity
- Track a user's password synchronization between directory environments

There are two different types of logs: error and audit.

- The error log includes entries about conditions qualified as severe errors and warnings. All error log entries are worthy of attention, and thus the administrator cannot prevent errors from being logged--if an error condition takes place, it will always be documented in the error log
- The audit log contains information about the day-to-day activities of the system. These include important events such as a user's password being synchronized between directories. The administrator can control the level of information that is logged, increasing or decreasing the detail of the log messages

Connectors

Directory-specific connectors are responsible for bi-directionally synchronizing password updates between directories/domains and includes

- **Directory Server connector.** It supports a single naming context (for example suffix/database) in a Sun One Directory Server 5.2

- **Active Directory connector.** It supports a single domain in a Windows 2000 Active Directory environment
- **NT connector.** It supports a single domain in Windows NT environment

Connector subcomponents

The Directory Server and the Windows NT connectors each make use of a lightweight subcomponent to facilitate the propagation of password updates. Connector subcomponents are installed along with the directory being synchronized and communicate with the connector over an encrypted connection.

The Active Directory connector does not require any subcomponents.

Directory Server subcomponent: plugin

A plugin is installed in the Directory Server being synchronized. The plugin,

- Enhances the Directory Server connector change detection features
- Provides bi-directional support for password synchronization support between Active Directory and Directory Server without requiring the installation of the Password Filter DLL in every Windows Domain Controller

NT connector subcomponents: Change Detector & Password Filter DLL

An installation supporting synchronization with NT SAM Registries requires two small processes to be installed in the Primary Domain Controller (PCD) along with the NT connector, they are:

- **The Change Detector.** It detects user entry and password change events by monitoring the Security Log, and then passes the changes to the connector
- **The Password Filter DLL.** It captures passwords in the clear at the NT Domain and passes these to the NT connector

Command Line Interfaces

There are a number of commands. The following are most notable towards the system configuration and towards the synchronization of passwords

- **`idsync linkusers`.** This command enables the administrator to link existing users in two directories. It is run after all connectors have been installed and while the system is not synchronizing users. This interface accepts rules for matching users between the two directories (e.g. for a user entry to be linked in the two directories both the first names and last names must match in both directory entries).
- **`idsync resync`**. During runtime, the product synchronizes user creations and modifications in real-time, but it does not bulk synchronize existing attributes that have not changed; this command line interface (after executing linkusers) can be used to synchronize existing attribute values between existing directory populations after linkusers has been run.
- **`idsync prepds`**. This command line interface must be run for every configured Directory Server master being synchronized; this step is a pre-requisite for installing the Directory Server Connector.

System Components Distribution

Before you can develop an effective deployment, you must understand how Identity Synchronization for Windows components are organized and how the product operates.

Once you understand the basic concepts described in this section and in the deployment scenario example, you should be able to extrapolate the information you need to create deployment strategies for more complex, sophisticated scenarios (such as mixed environments or multi-server environments).

Core

All core components are installed at once in any of the supported OS platforms; Administration Server must be installed in the same machine as core.

Directory Server connector

Directory Server connectors can be installed in any of the supported OS platforms; there is no requirement for installation of a Directory Server connector in the same machine where the Directory Server being synchronized is running. There must be one Directory Server connector installed per Directory Server’s naming context/(database/suffix.)

Directory Server (subcomponent) plugin

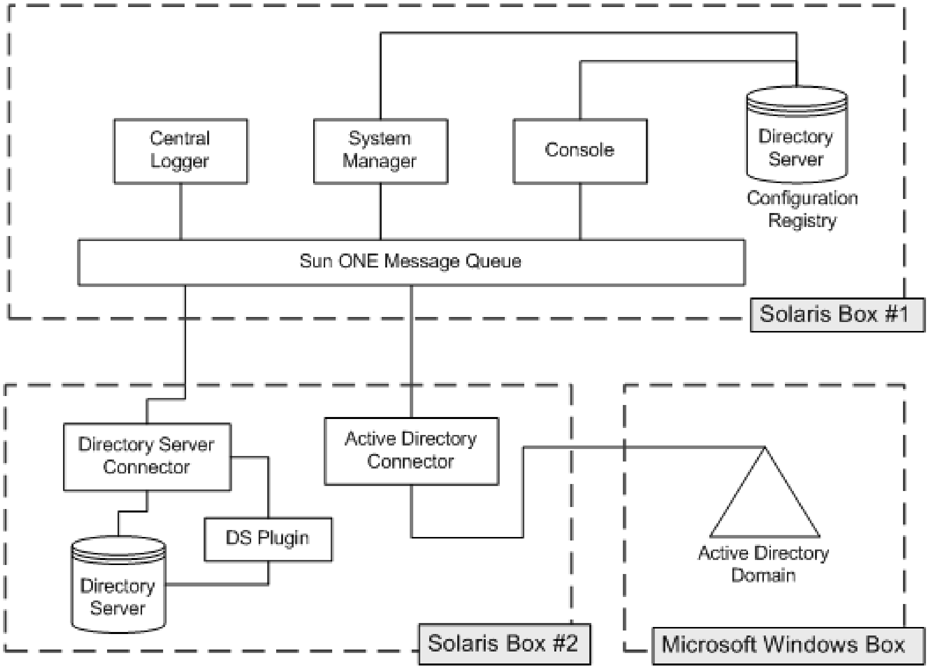
The Directory Server plugin is installed in some host where the Directory Server being synchronized resides on any of the supported OS Platforms.

NOTE A single Directory Server connector is installed for a single Directory Server naming context. However, Directory server plugins are installed multiple times for every master, hub, and read-only replica for that naming context (suffix-database.)

Active Directory Connector

Active Directory connectors can be installed in any of the supported OS platforms; there is no requirement for installation of an Active Directory connector in the Windows environment. There must be one Active Directory connector installed per Active Directory domain. There is no requirement to install any subcomponents for this connector type.

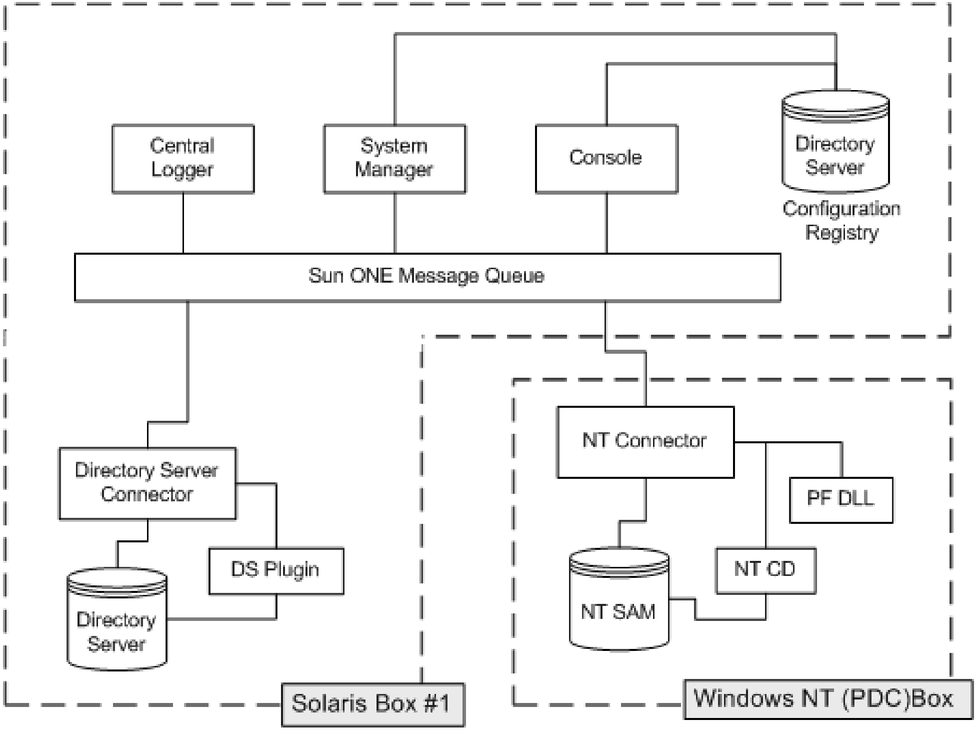
Figure 1-2 Directory Server and Active Directory Component Distribution



NT Connector & NT subcomponent

Installations that support synchronization with NT SAM Registries (seeFigure 1-3) require the NT connector to be installed in the Primary Domain Controller (PDC); in addition, the two NT connector subcomponents, the Change Detector and the Password Filter DLL, must also be installed along with the connector in the PDC of the NT Domain. A single NT connector synchronizes passwords for a single NT Domain.

Figure 1-3 Directory Server and NT Component Distribution



Deployment Example: A Two-Machine Configuration

This section describes a deployment scenario in which Identity Synchronization for Windows is used to synchronize creations and bidirectional password modifications between Sun and Windows directories.

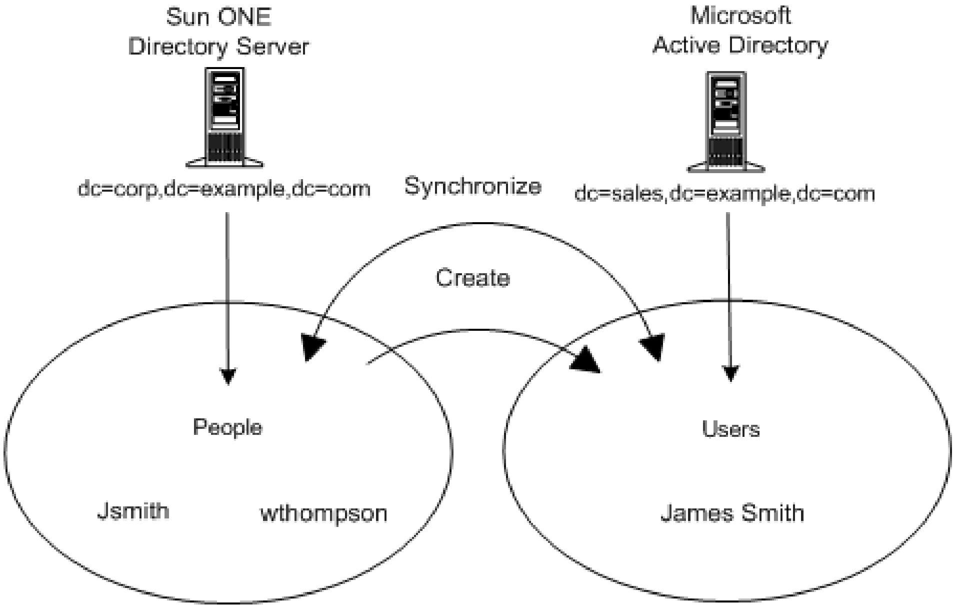
The deployment scenario consists of two systems:

- A system running a Sun ONE Directory Server (host name: *corp.example.com*)
- A system running Active Directory on a Windows 2000 server (host name: *sales.example.com*)

NOTE Though NT is not used in this scenario, it is important to note that Identity Synchronization for Windows also supports synchronization with NT domains.

The following figure illustrates the synchronization requirements (node structures with associated attribute values) used for this deployment scenario.

Figure 1-4 Synchronization Requirements



There are two goals for this scenario:

- The first goal is to synchronize user passwords bidirectionally between the *user subtrees* (*ou=people* in Directory Server and *cn=users* in Active Directory), which means that whenever a user password changes in either directory, the password change "flows" to the associated user in the other directory.

  For example, if you change the password for uid=JSmith in the ou=people container on the Sun ONE Directory Server, then the new password should automatically flow to cn=Joe Smith in the cn=users container on the Active Directory server and vice versa.

- The second goal is to synchronize or propagate user creations from the Directory Server user subtree to the Active Directory user subtree only.

  For example, if you create a new user (uid=WThompson in the ou=People container) with a specified set of attributes and then create an account for WThompson (cn=William Thompson in the cn=Users container) with the same set of attributes.

NOTE Identity Synchronization for Windows supports multiple synchronization sources of the same type (for example, you can have more than one Sun ONE Directory Server in a deployment or multiple Active Directory Domains).

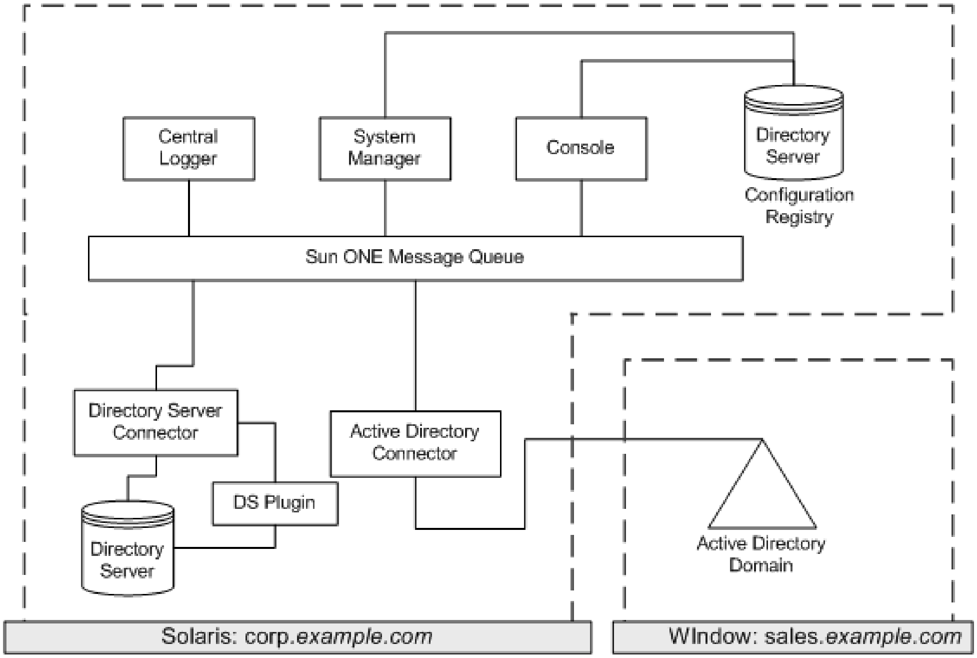
It is important to note that creation and modification synchronization settings are global for the entire set of directories, and cannot be specified for individual directory sources.

NOTE For example, if settings are set to synchronize user creations from Sun to Windows, then user creations propagate from all Sun Directory Servers to all Windows Active Directory domains and Windows NT domains configured in the installation.

Physical Deployment

Figure 1-5 illustrates how all the product's components are physically deployed in a single Solaris box while the Active Directory domain resides in a separate Windows 2000 Active Directory domain controller where no components have been installed.

Figure 1-5 Directory Server and Active Directory Scenario



Component Distribution

Host ds.example.com is a Directory Server box installed in a Solaris operating environment. This machine contains:

Directory Server being synchronized, its naming context (Root Suffix) is dc=corp,dc=example,dc=com

- Identity Synchronization for Windows core components
- Identity Synchronization for Windows Directory Server connector
- Identity Synchronization for Windows Directory Server (subcomponent) plugin
- Identity Synchronization for Windows configuration registry (in a separate Directory Server than the one being synchronized, a Configuration Directory)

Host ad.example.com is an Active Directory domain controller named sales.example.com that contains an Active Directory domain being synchronized, its domain name is sales.

Chapter 2

Preparing for Installation

Before installing Sun ONE Identity Synchronization for Windows, familiarize yourself with the installation and configuration process.

This chapter contains the following sections:

- Installation Overview
- Configuration Overview
- Installation and Configuration Decisions
- Installation Checklists
- Installation Requirements
- Unpacking the Software
- Installation Privileges

Installation Overview

Identity Synchronization for Windows is made out of two significant types of components: core and connectors. The order by which these components are installed and configured is as follows:

1. Core is installed using the installer
2. The product is initially configured via the console
3. Once configuration is complete, any number of connectors and subcomponents are installed using the installer; the number of connector/subcomponent installs depends on the number of directories to be synchronized.

Core Installation

When core installation is complete the following components will have been installed:

- Sun ONE Message Queue
- Console
- Configuration Registry
- Central Logger
- System Manager

Deployment Configuration

Using the console or the equivalent command line interface, the Administrator initially configures the Directory Sources to be synchronized, and other characteristics of the deployment, all from a centralized location. See Configuration Overview in this Chapter.

Prepare Directory Server Command Line Interface

Directory Server connector supports the Sun ONE Directory Server 5.2 naming context (suffix/database.) The command `prepds` must be run for every master (preferred secondary) Directory Server of the being synchronized

NOTE	Running `prepds` is a pre-requisite for installing a Directory Server Connector for a Directory Server being synchronized.

Connector & Connector Subcomponent Installation

The number of Connector and subcomponent installs depends on the type and number of configured directories.

NOTE The console and the installer associate a directory being synchronized and its connector by the directory's label, see table below for label naming conventions.

Table 2-1 Label Naming Conventions

Connector Type	Directory Source Label	Subcomponent?
Directory Server connector	Naming context or suffix/database	Directory Server Plugin: 1 to n required. Install one in every Directory Server (master or consumer) for the naming context being synchronized
AD connector	Domain name	None
NT connector	Domain name	Change Detector and PF DLL:1 set required. Install each pair for every NT connector; these subcomponents are installed together in the same installation.

Optional Installation Steps

linkusers Command Line Interface

This bootstrap phase is a bulk load process by which the entries in both the Windows and Sun ONE Directory Server directories are uniquely identified and linked each other; it is run after all connectors have been installed and have reached the READY state (the connector's state is visible via the console Status panel.

resync Command Line Interface

During runtime, the product synchronizes user creations and modifications in real-time, but it does not bulk synchronize existing attributes that have not changed; therefore, `resync` can be used to synchronize existing attribute values between existing Sun and Windows Directory Source populations.

Configuration Overview

Once you have installed core, the next step is to configure the product deployment, which involves configuring directories to be synchronized and synchronization settings for attribute modifications and optionally user entry creations between the configured directories.

Please familiarize yourself with the following configuration element concepts:

- Synchronization Settings
- Directories
- Global Catalog and Configuration Directory
- User Objectclass
- Significant Attributes
- Creation Attributes
- Attribute Maps
- Synchronization User Lists

Synchronization Settings

These settings specify in which direction to synchronize user creations and attribute level modifications between Sun and Windows directories. Possible settings are:

- Sun to Windows
- Windows to Sun
- Bi-directional

NOTE It is not possible in a Sun-Active Directory / NT configuration to save a configuration that specifies different synchronization settings for creations or modifies from NT to Sun and from Active Directory to Sun

Directories

A directory represents:

- A single naming context (suffix/database) in one or more Sun ONE Directory Server 5.2
- A single Active Directory Domain in a Windows 2000 Active Directory Forest
- A single NT Domain

You might configure any number of directories of each type.

Global Catalog and Configuration Directory

Identity Synchronization for Windows uses these repositories to fetch the Active Directory or Directory Server directory topology, as well as the schema information for the directories.

User Objectclass

This is the objectclass of the user entry in the Sun or in the Active Directory space (inetorgperson, User, etc.).

NOTE The user objectclass is configurable for Directory Servers but not so for Active Directory (defaults to User) or not applicable for NT.

Significant Attributes

Significant attributes are synchronized in addition to passwords; these attributes are synchronized between the Sun and Windows directories every time they are modified according to the modification synchronization settings.

Creation Attributes

Creation attributes are synchronized in addition to passwords; these attributes are synchronized whenever a new user is created at either the Sun or the Windows directories depending on to the creation synchronization settings.

NOTE Please note that significant attributes are automatically synchronized as creation attributes but not the other way around, creation attributes are only synchronized during user creations.

Mandatory Creation Attributes are attributes considered “mandatory” in order to successfully complete a creation action on the target directory. For example, Active Directory expects that both cn and samaccountname have valid values upon user creation, On the Sun side, if configuring inetorgperson as the user objectclass then cn and sn will be expected as mandatory attributes for a creation. You must provide attribute maps for mandatory creation attributes.

A creation attribute default updates the target directory creation attribute with a default value ONLY when there is no value in the attribute propagated from the originating directory.

Attribute Maps

An attribute map maps the name of the Sun attribute to the name of the Windows attribute and vice versa.

- Attribute maps are used for both Significant and Creation Attributes
- Attribute maps must be configured for all “mandatory creation attributes” in each directory type

Synchronization User Lists

A Synchronization User List defines which specific users in two directories are to be synchronized; these definitions enable synchronization of a flat DIT to a hierarchical DIT:

- Which users will be synchronized?
- Which users are excluded from synchronization?
- Where should new users be created?

A Synchronization User List includes two definitions; each definition identifies the group of users to be synchronized in the topology terms of the directory type.

- One identifies Directory Server users to synchronize (for example: ou=people, dc=example, dc=com)

- The other one identifies Windows users to synchronize (for example: cn=users, dc=example, dc=com)

The following concepts are used to define a Synchronization User List:

- The Synchronization User List Base DN includes all users under that DN unless another Sync Scope is more specific or unless excluded by the filter
- A Synchronization User List's filter uses attributes in the user's entry to exclude users from synchronization or to separate users with the same base DN into multiple Sync Scopes
- A Synchronization User List's creation expression constructs the DN where new users are created, "cn=%cn%,ou=sales,dc=sun,dc=com" where %attr% is replaced with the value from the user entry. A creation expression must end with the base DN.

See Synchronization User List Definitions and Configuration for detailed information about Synchronization User Lists.

Installation and Configuration Decisions

This section gives installation and configuration summaries and details the choices you make in deploying Identity Synchronization for Windows. Have this information available before you begin the installation process. This section contains:

- Installation Summary
- Configuration Summary
- Core Installation
- Core Configuration
- Connector and Subcomponent Installation
- Optional Command Line Interface Usage

Installation Summary

To install Identity Synchronization for Windows perform the following steps:

- Install core
- Configure the product via console

- Run command `idsync prepds` for every Directory Server being synchronized
- Install a Directory Server connector for every Directory Server master being synchronized
- Install a Directory Server (subcomponent) plugin for every Directory Server master, hub and read-only replica that stores users being synchronized
- Install an Active Directory connector for every Active Directory domain being synchronized (if synchronizing between Sun and Windows 2000)
- Install an NT connector and NT subcomponents for every Active Directory domain being synchronized (if synchronizing between Sun and Windows NT)
- Run (optional) commands `idsync linkusers` and `resync`
- Start synchronization

Configuration Summary

During installation, you are prompted for basic configuration information. Decide how you are going to configure these basic parameters before you begin the installation process.

1. Open console
2. Configure directories to be synchronized
3. Configure Active Directory Global Catalog (mandatory in Sun / Active Directory synchronization) and Configuration Directory (optional, it is generally automatically configured by console)
4. Configure modification synchronization settings
5. Select Sun-side user objectclass
6. Optionally add and map significant attributes to synchronize (in addition to passwords)
7. Optionally configure user creation synchronization settings
8. Configure creation mandatory and optional attributes and maps
9. Configure synchronization user lists
10. Save configuration
11. Proceed with connector and subcomponent installation

Core Installation

- **JAVA_HOME** must be set on all systems. or best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4.1-03 or higher and not the Java Runtime Environment (JRE) is available on the host.
- **Configuration Directory URL.** The configuration directory is the Directory Server instance where Identity Synchronization for Windows configuration information is to be stored.
- **Root suffix** for the configuration directory. All configuration information is stored under this suffix.
- **File system directory** location to install Identity Synchronization for Windows. Core must be installed in the same directory as a Directory Server Administration server.
- Configuration Directory Server administrator's **name and password**.
- A secure **configuration password** that is used to protect sensitive configuration information.
- **An unused port number** for the Sun ONE Message Queue instance.

Core Configuration

- **Sun ONE Directory schema server.** The desired Directory Server data loaded from the Configuration Directory.
- **User object class (for Directory Server only).** The user object class is used to determine user types. Based on this object class a list of attributes, including password attributes, is derived. This list is populated from the schema.
- **Synchronized Attributes.** User entry attributes that ought to be synchronized between the Sun ONE Directory Server and the Windows environment.
- **Attribute Modifications flow.** Decide whether you wish user changes made in the Sun ONE Directory Server environment to propagate to Windows servers and/or changes made in the Windows environment to propagate to the Sun ONE Directory Server.
- **Creation flow.** Decide whether you wish user creations made in the Sun ONE Directory Server environment to propagate to Windows environments and/or creations made in the Windows environment to propagate to the Sun ONE Directory Server.

- **Directory sources.** These represent the location of user information such as Active Directory and Sun ONE Directory Server.
- **Global Catalogs.** These are repositories of Windows topological and schema information.
- **Active Directory schema controller.** The Fully Qualified Domain Name (FQDN) of the desired Active Directory schema source retrieved from the Windows global catalog.
- **Configuration Directory.** The Sun ONE Directory Server storing the Identity Synchronization for Windows configuration.
- **Active Directory Source.** These are to synchronize the Active Directory domains.
- **Windows NT Primary Domain Controller.** Know the names of the NT domains to be synchronized and the names of the Primary Domain Controller for each domain.
- **Synchronization User Lists.** Identify the sets of users to be synchronized on both the Sun ONE Directory Server, Active Directory, and NT.

Connector and Subcomponent Installation

- **JAVA_HOME** must be set on all systems. or best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4.1-03 or higher and not the Java Runtime Environment (JRE) is available on the host on which connector installation is being performed.
- **Configuration Directory URL.** The Sun ONE Directory Server instance where Identity Synchronization for Windows configuration information is stored.
- **Root suffix** for the Configuration Directory.
- The secure **configuration password** that was chosen during core installation.
- **File system directory** in which to install the connector.
- Configuration Directory Server **administrator name and password**.
- **Directory sources.** Identity Synchronization for Windows uses connectors to synchronize user passwords between directory sources. This is the directory source with which you wish to connect.

Optional Command Line Interface Usage

Several command line interface scripts are available to deal with existing user populations. This section describes their usage:

- `linkusers`
- `resync`
- Post-installation Recommendations

`linkusers`

The bootstrap phase differs if the Sun Directory Server is empty or already populated, and if existing Directory Server entries are already linked to their counterparts in the Windows environment. The assumption is that at a minimum, prior to running this command, Active Directory will already have been populated and the users selected that require password synchronization. Table 2-2 summaries these conditions.

Table 2-2 Existing Conditions

Existing condition	comments
Active Directory -only populated	Directory Server has been installed (it may even be in use for other purposes) but it does not hold any user entries or directory structure that mirrors the Active Directory user entries.
Directory Server and Active Directory populated but not linked	Directory Server and Active Directory both hold overlapping user entries and directory structures that are targeted for password synchronization. However, the two user repositories have not been linked.

`resync`

Table 2-3 provides examples for deciding when to use the `resync` command.

Table 2-3 `idsync resync` Usage

Existing Need	When to use resync
Initially synchronize existing Sun and Windows Directory Source populations	Use `resync` to bulk resynchronize: synchronize existing attribute values between existing Sun and Windows Directory Source populations

Existing Need	When to use resync
	`resync` must always be run in a deployment with existing Window's users before starting synchronization for the first time.
Populate empty Active Directory, NT or Directory Server	Use `resync` to populate an empty Directory Server with users from Active Directory or Windows NT.
	Use `resync` to populate an empty Directory Server with existing Active Directory or Windows NT users
Re- synchronize user entries after a failure or out of sync condition	If two directory sources become out of sync, then `resync` can once synchronize user entries.
Prime Windows-side Object Cache	`resync` can "prime" the object cache database of the NT and AD connectors. The object cache maintains a shadow copy of the AD or NT SAM. When the product is first installed, `resync` primes the object cache to match the contents of the AD or NT SAM.
Create users	`resync` can create users and synchronize attributes, but it cannot synchronize passwords. However, `resync` can synchronize existing Active Directory passwords with their corresponding Directory Server entry.
	When running `resync` with the `-i ALL_USERS` option, on-demand password synchronization occurs between each Active Directory user entry in the Directory Server.

Post-installation Recommendations

If users exist in the Windows directories, then the `resync` command must be run before synchronization is started.

- If you do not want to synchronize existing users to the Sun directories, then run it with the -u flag, which only updates the object cache and does not synchronize the Windows entries to Directory Server.
- If you have existing Windows users, and you do not run `resync` before starting synchronization for the first time, then changes to these users may or may not be propagated, and depending on flow settings, they might even be automatically created in Directory Server. `resync` must be run even if `linkusers` was already run.

The following table summarizes the post-installation steps that must be followed based on existing user populations.

Table 2-4 Post Installation Steps

Users Exist In		**Required Post-installation Steps to Follow**	
Windows	Sun	Existing users should be synchronized	Existing users should NOT be synchronized
No	No	None	None
No	Yes	Run `idsync resync -o Sun -c` to create existing Sun directory users in Windows	None
Yes	No	Run `idsync resync -c` to create existing Windows users in the Sun directory	Run `idsync resync -u` to populate the connector's local cache of user entries.
Yes	Yes	Run `idsync linkusers` to link users between Windows and Sun. Then run `idsync resync` or `idsync resync -o Sun` to synchronize existing user values between the two directories.	Run `idsync resync -u` to populate the connector's local cache of user entries.

Installation Checklists

These checklists are intended to aid in the installation process. Print them out and record the following information prior to installing Identity Synchronization for Windows.

Core Installation

Required Information	Entry
JAVA_HOME must be set on all systems. or best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4.1-03 or higher and not the Java Runtime Environment (JRE).	
Configuration Directory URL.	
Root suffix for the configuration directory such as dc=example,dc=com.	
File system directory in which to install Identity Synchronization for Windows.	
Configuration Directory Server administrator's name and password.	
A secure configuration password that is used to protect sensitive configuration information.	

Required Information	Entry
The port number of the Sun ONE Message Queue instance.	

Core Configuration

Required Information	Entry
Active Directory Global Catalog when appropriate.	
Sun ONE Directory Server schema server.	
Sun ONE Directory Server User object class.	
Synchronized Attributes.	
Flow of user modifications.	
Flow of user creations.	
Sun ONE Directory sources.	
Active Directory Sources.	
Synchronization User Lists.	
Windows source filter creation expression	

Required Information	Entry
Sun ONE source filter creation expression	

Connector and Subcomponent Installation

Required Information	Entry
Configuration Directory URL.	
Root suffix for the configuration directory.	
File system directory in which to install the connector.	
Configuration Directory Server administrator name and password.	
A secure configuration password that is used to protect sensitive configuration information.	
JJAVA_HOME must be set on all systems. or best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4.1-03 or higher and not the Java Runtime Environment (JRE) is available on the host.	
Directory sources.	

Linking Users

Required Information	Entry
The synchronization user lists to be linked.	
The attributes that are used to match equivalent users.	
XML configuration file.	

Resynchronization

Required Information	Entry
SUL selection.	
Synchronization source.	
Whether to automatically the create a user entry, if a corresponding user is not found at the destination directory source.	
Whether or not Sun ONE Directory Server passwords should be invalidated.	
Whether only users that match the specified ldap filter and are in the selected SULs will be synchronized.	

Installation Requirements

This section covers the required operating system version, patches, and utilities for each platform. The following hardware and software are required for this release of Identity Synchronization for Windows.

Table 2-5 Hardware and Software Requirements

Component	Solaris Requirement	Windows Requirement
Core	Sun Solaris 8 for UltraSPARC (32 and 64 bit) Sun Solaris 9 for SPARC® platforms (32 and 64 bit)	Windows 2000 Server SP4 Windows 2000 Advanced Server SP4
Sun ONE Directory Server and Active Directory connectors	Sun Solaris 8 for UltraSPARC (32 and 64 bit) Sun Solaris 9 for SPARC® platforms (32 and 64 bit)	Windows 2000 Server SP4 Windows 2000 Advanced Server SP 4
Sun ONE Directory Server plug-in	Sun Solaris 8 for UltraSPARC (32 and 64 bit) Sun Solaris 9 for SPARC® platforms (32 and 64 bit)	Windows 2000 Server SP4 Windows 2000 Advanced Server SP 4
NT connectors and subcomponents		Windows Primary Domain Controller NT 4.0 Server SP 6A (x86 only)

Sun ONE Software Requirements

The following Sun ONE software components must be installed:

- Sun ONE Directory Server version 5.2 or higher

  For the latest information about patches that may be required to install Directory Server 5.2 on Solaris, refer to the *Sun ONE Directory Server 5.2 Installation and Tuning Guide* and the *Sun ONE Directory Server 5.2 Release Notes*, which can be found at the following web site:

  `http://docs.sun.com/db/coll/S1_DirectoryServer_52`

- Sun ONE Message Queue version 3.0.1 (Installed with Identity Synchronization for Windows if not already present).

- JAVA_HOME must be set to a 1.4.1_03 JRE or later on Solaris before installation. If this is not done, the installer might report `JAVA_HOME not set.`

Hardware Requirements

On all platforms, you will need:

- Roughly 400 MB of disk space for a minimal installation on Directory Server.
- 512 MB minimum of RAM on the Core server; 1 GB preferred.

Configuring Windows for SSL Operation

If you are planning to propagate password changes from Sun ONE Directory Server to Windows Active Directory servers you must configure each Active Directory server to use SSL.

The Identity Synchronization for Windows Active Directory connector installer can automatically setup SSL in the Active Directory connector if LDAP over SSL in AD has been enabled by automatically obtaining a certificate from a Microsoft Certificate Services Enterprise Root certificate authority as described in:

`http://support.microsoft.com/default.aspx?scid=kb;en-us;q247078.`

However, LDAP over SSL can more easily be configured as described in this MSDN tech note:

`http://support.microsoft.com/default.aspx?scid=kb;en-us;321051.`

In this case, the administrator must manually install the certificate in the connector's certificate database as described in Enabling SSL in the Active Directory Connector.

Unpacking the Software

If you have downloaded Identity Synchronization for Windows software, unpack it before beginning installation.

1. Create a new directory for the installation:

```
# mkdir pwsync
```

```
# cd pwsync
```

2. Download the product binaries file to the installation directory.
3. On a UNIX system, unpack the product binaries file using the following command:

   ```
   # gzip -dc file_name.tar.gz | tar -xvof -
   ```

   where *file_name* corresponds to the product binaries that you want to unpack.

   On a Windows NT system, unzip the product binaries.

Installation Privileges

On UNIX systems, you must install as root.

On Windows systems, you must run the installation as Administrator.

Chapter 3

Core Installation

This section contains procedures for using the Identity Synchronization for Windows setup program. Perform the following procedures in the order presented:

- Starting the Installer
- Core Installation

Starting the Installer

To prepare and start the installer perform the following steps:

1. Log in as root or an Administrator as appropriate.

NOTE If the Identity Synchronization for Windows has been previously installed, remove it following the procedures found at "Removing the Software" on page 119.

2. Create a new directory:

```
# mkdir isw10
# cd isw10
```

3. If you have not already done so, download the product binaries file to the installation directory.

4. Unpack the product binaries file using the following command in the UNIX environment:

```
# gunzip -dc file_name.tar.gz | tar -xvof -
```

where *file_name* corresponds to the product binaries that you want to unpack.

5. On a Windows system unzip the product binaries file.

6. Run the installer. You can find it in the directory where you untarred binary files. On Windows machines execute the setup executable in the installer directory:

   ```
   cd installer
   ```

   ```
   setup.exe
   ```

   Execute the installer on UNIX machines:

   ```
   ./runInstaller.sh
   ```

NOTE JAVA_HOME must be set on all systems. or best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4 or higher and not the Java Runtime Environment (JRE) is available on the host on which connector installation is being performed.

NOTE To run the Installer in text based mode enter:

```
./runInstaller.sh -nodisplay
```

Core Installation

Install the Identity Synchronization for Windows Core components by performing the following steps:

NOTE Identity Synchronization for Windows 1.0 requires root privileges on Solaris to install and run its services. If you wish to run services on Solaris under a non-root user see“Running Services as Non-Root” on page 211.

Installation of Identity Synchronization for Windows 1.0 on Windows 2000 requires logging on as Administrator.

1. At the welcome screen press Next.

2. At the Software License Agreement, read the license screen and press Yes (Accept License) to accept the license terms. Press No to exit setup.

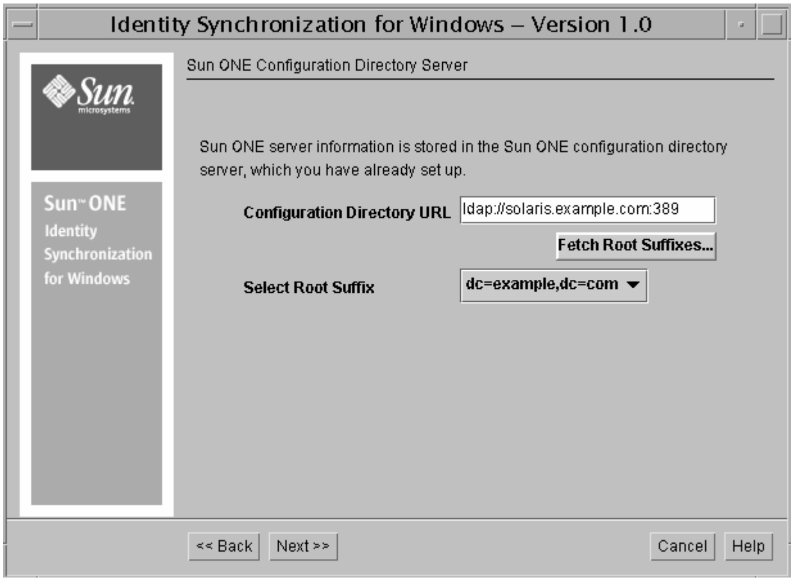
3. When prompted, enter the Configuration Directory URL.

The configuration directory is the fully qualified domain name of the Directory Server instance where configuration information is to be stored. Enter the following:

ldap://*Directory Server name*:*port number*

NOTE Host names should be DNS resolvable to the machine on which the console is running in order to avoid warnings that credentials and /or host names could not be validated.

4. Select the root suffix where you would like the Identity Synchronization for Windows configuration to be stored. Press Fetch Root Suffixes and a drop-down list will populate with choices. Select the desired root suffix.



Identity Synchronization for Windows – Version 1.0

Sun ONE Configuration Directory Server

Sun ONE server information is stored in the Sun ONE configuration directory server, which you have already set up.

Configuration Directory URL: ldap://solaris.example.com:389

Select Root Suffix: dc=example,dc=com

5. Press Next.

6. Enter the Configuration Directory Server administrator’s name and password and press Next.

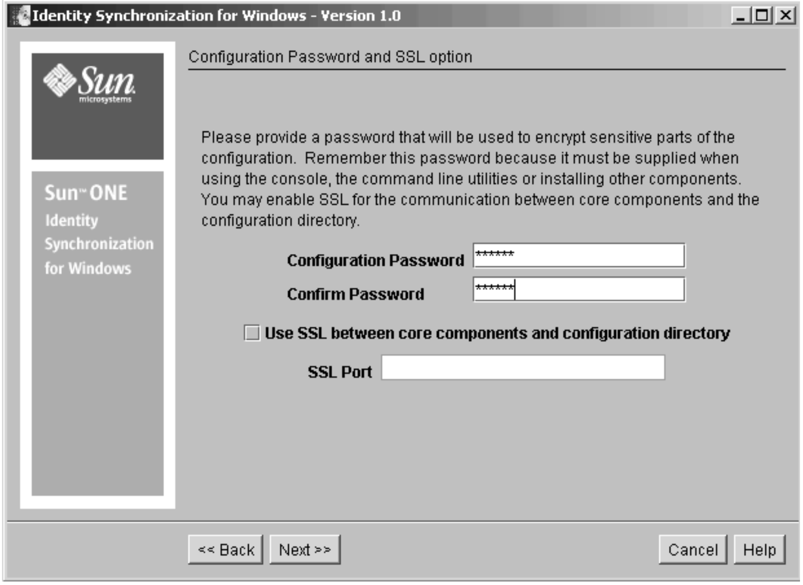
NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

7. Enter and confirm a password used to encrypt sensitive configuration information.

NOTE Remember this password as it is used to configure the system and to access the console. For information on changing the configuration password see "Using changepw" on page 186.

If desired, select Use SSL between core components and configuration directory and provide the SSL port. Press Next.

NOTE Sensitive configuration information is encrypted before it sent to the configuration Directory Server. If however, further transport encryption is desired between the console and configuration directory, first make sure SSL has been enabled for both the Sun ONE Administration Server and Configuration Directory Server. Then, you should set up a secure connection between the Administration Server to which you will be authenticating the Sun ONE Server Console (see Administration Server Configuration: The Configuration Directory in the *Server Management Guide*).



Identity Synchronization for Windows - Version 1.0

Configuration Password and SSL option

Please provide a password that will be used to encrypt sensitive parts of the configuration. Remember this password because it must be supplied when using the console, the command line utilities or installing other components. You may enable SSL for the communication between core components and the configuration directory.

Configuration Password: *******

Confirm Password: ******

□ Use SSL between core components and configuration directory

SSL Port:

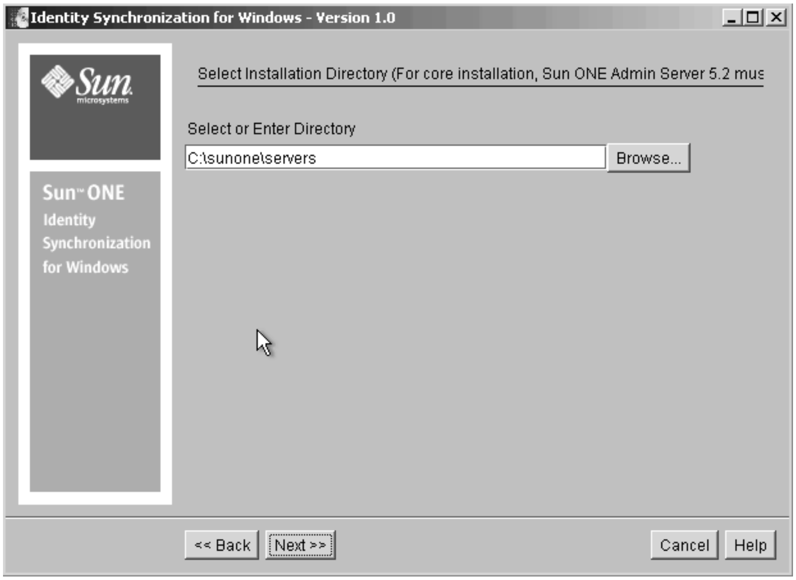
8. Enter the directory to install Identity Synchronization for Windows. Press Browse to select available directories. For example, on a Windows machine enter:

   ```
   C:\Program Files\Sun\MPS
   ```

   On a UNIX machine enter:

   ```
   /var/Sun/mps
   ```

NOTE The file system directory where core is installed must be an existing server root managed by an Administration Server version 5.2. The installer will not install core into a file system directory that does not meet this criteria. Administration Server 5.2 can be installed through the Sun ONE Directory Server 5.2's installation program.



Identity Synchronization for Windows - Version 1.0

Select Installation Directory (For core installation, Sun ONE Admin Server 5.2 mus

Select or Enter Directory: C:\sunone\servers

9. Press Next.

10. You are then prompted Do you want to install a new Sun ONE Message Queue? Click No to use an existing version of Sun ONE Message Queue. Click yes if there is not an existing instance of Sun ONE Message Queue.

NOTE An administrator user is created by core installer for managing Sun ONE Message Queue. This user has administrator privileges for Sun ONE Message Queue. This password is set to the configuration password during core installation. If you run `idsync changepw`, this does not change this password. Of course, the Directory Manager can change the password as desired. The entry is

```
cn=administrator,ou=MessageQueueCredentials,
ou=GlobalConfig,ou=1.0,ou=IdentitySynchronization,
ou=Services,<configuration - root suffix>
```

NOTE If you desire to use an existing Sun ONE Message Queue, it can not be currently in use in any way. If you choose an existing SUN ONE Message Queue and it is already configured and in use, the Message Queue Broker configuration will be over written.

There can only be one Message Queue Broker instance running per host.

An existing Sun ONE Message Queue must be version 3.0.1 sp2 or greater.

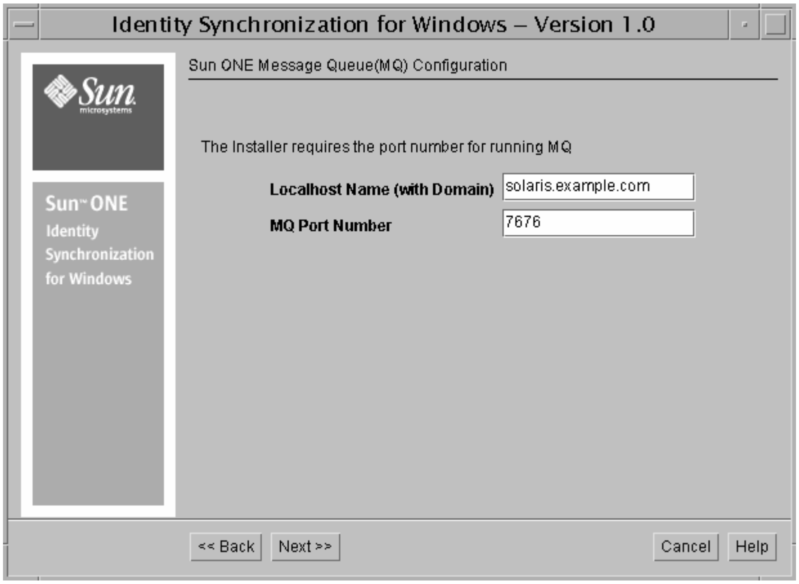
11. Click Next.

12. Enter the Localhost Name and Port Number of the Sun ONE Message Queue; for example:

`hostname``.example.com`

`7676`

NOTE On Solaris systems Sun ONE Message Queue should not be installed in the same directory as Identity Synchronization for Windows


Identity Synchronization for Windows – Version 1.0
Sun microsystems
Sun™ ONE Identity Synchronization for Windows
<< Back
Next >>
Cancel
Help

Sun ONE Message Queue(MQ) Configuration

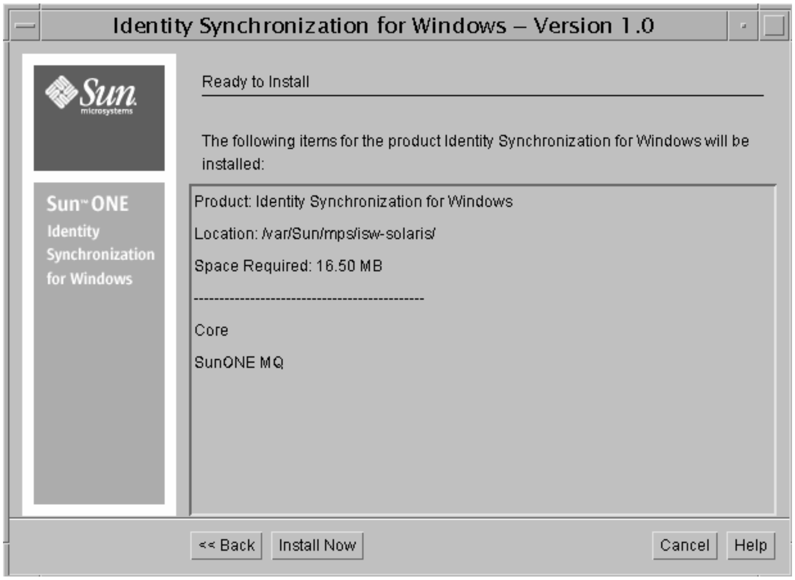
The Installer requires the port number for running MQ

Localhost Name (with Domain) solaris.example.com

MQ Port Number 7676

13. Click Next.

The setup program checks for available disk space and a installation summary menu appears.


Identity Synchronization for Windows – Version 1.0
Sun microsystems
Sun™ ONE Identity Synchronization for Windows
<< Back
Install Now
Cancel
Help

Ready to Install

The following items for the product Identity Synchronization for Windows will be installed:

Product: Identity Synchronization for Windows

Location: /var/Sun/mps/isw-solaris/

Space Required: 16.50 MB

Core

SunONE MQ

14. Ensure that the following components are loading into the desired directory:

    ```
    Core
    ```

    ```
    SunONE MQ
    ```

    When ready, click Install Now.

15. Setup then installs Identity Synchronization for Windows.

    The Register Configuration Data menu appears.

16. Click Next.

    Setup enables and configures Sun ONE Message Queue. This can take several minutes.

17. An Installation Summary appears. Press Details if you wish to view the installation log. Press close to exit setup.

    A message appears asking if you wish to start the console.

18. Configure the Identity Synchronization for Windows core by following the procedures found in the Chapter 4, “Resource Configuration”.

Chapter 4

Resource Configuration

This chapter provides procedures for configuring the Identity Synchronization for Windows core server using the console. Perform these procedures immediately after installing the Identity Synchronization for Windows core as described in Chapter 3, "Core Installation". Use these procedures to add or modify resources within your network.

NOTE Core configuration may also be accomplished by using the `idsync importcnf` command. For information on creating XML configuration files and importing configurations see the "Using importcnf" on page 187.

Upon completion of initial core configuration, use the setup program to install Identity Synchronization for Windows connectors following procedures found in the Chapter 5, "Connector and Subcomponent Installation".

The procedures in this section presume that the administrator is knowledgeable about Sun ONE Directory Server and Active Directory configuration and operation.

This chapter contains the following sections:

- Initial Core Configuration
- Prepare Directory Server
- Continuing the Installation

Initial Core Configuration

All initial configuration steps use the Identity Synchronization for Windows console. Perform these steps to initially configure the Identity Synchronization for Windows core:

- Open the Appropriate Identity Synchronization for Windows Console
- Creating Directory Sources
- Deleting Directory Sources
- Setting Attribute Modification Flow
- Setting the Modification Attribute Mapping
- Setting Object Creation Flow and Attribute Mapping
- Creating Synchronization User Lists

Open the Appropriate Identity Synchronization for Windows Console

Access the Identity Synchronization for Windows Console following these procedures:

1. Open the Sun ONE Console application by using the appropriate option:

   ❍ For local access on a UNIX machine, at the command-line prompt, enter the following line: `<`*`server-root`*`>/startconsole`.

   ❍ For local access on a Windows 2000 or NT machine, double-click the Sun ONE Console icon typically found at: `C:\Program Files\Sun\MPS`.

   The Sun ONE Console Login window appears.

2. Authenticate yourself to the configuration directory.

   ❍ **User ID.** Type the *administrator ID* you specified when you installed Administration Server on your machine. You installed Administration Server when you installed your first Sun ONE server or as a part of Identity Synchronization for Windows installation.

   ❍ **Password.** Type the *administrator password* that you specified when you installed Administration Server on your computer.

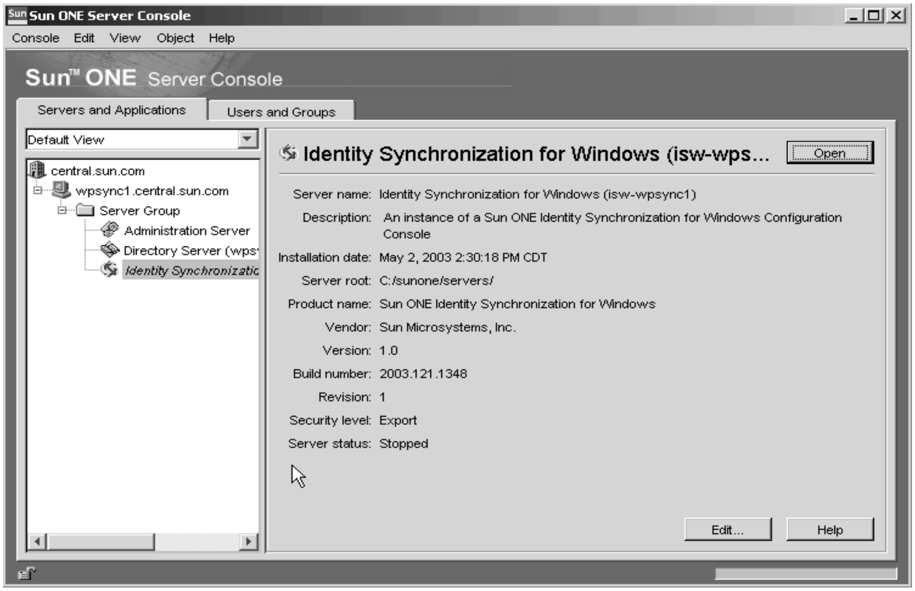
- **Administration URL.** This field should show the URL of the Administration Server. If it doesn't or if it doesn't have the URL of Administration Server that you want, type the URL in this field. The URL is based on the computer host name and the Administration Server port number you chose when you installed Administration Server. Use this format:

  `http://`*`hostname.your_domain.domain`*`:`*`port_number`*

  For example, if your domain name is `example.com` and you installed Administration Server on a host machine called `myHost` and specified port number `390`, the URL would look like this:
  `http://myHost.example.com:390`

3. Click OK.

   The Sun ONE Console appears with a list of all the servers and resources under your control.



Identity Synchronization for Windows (isw-wps...

Server name: Identity Synchronization for Windows (isw-wpsync1)
Description: An instance of a Sun ONE Identity Synchronization for Windows Configuration Console
Installation date: May 2, 2003 2:30:18 PM CDT
Server root: C:/sunone/servers/
Product name: Sun ONE Identity Synchronization for Windows
Vendor: Sun Microsystems, Inc.
Version: 1.0
Build number: 2003.121.1348
Revision: 1
Security level: Export
Server status: Stopped

4. Expand the hostname that contains the server group to which the Identity Synchronization for Windows instance belongs.

5. Expand the Server Group node, select the entry that corresponds to the desired Identity Synchronization for Windows instance, and click Open.

6. Enter the configuration password that was specified during core installation. (See Core Installation).

   The Identity Synchronization for Windows Console opens.

7. Press the Configuration tab.

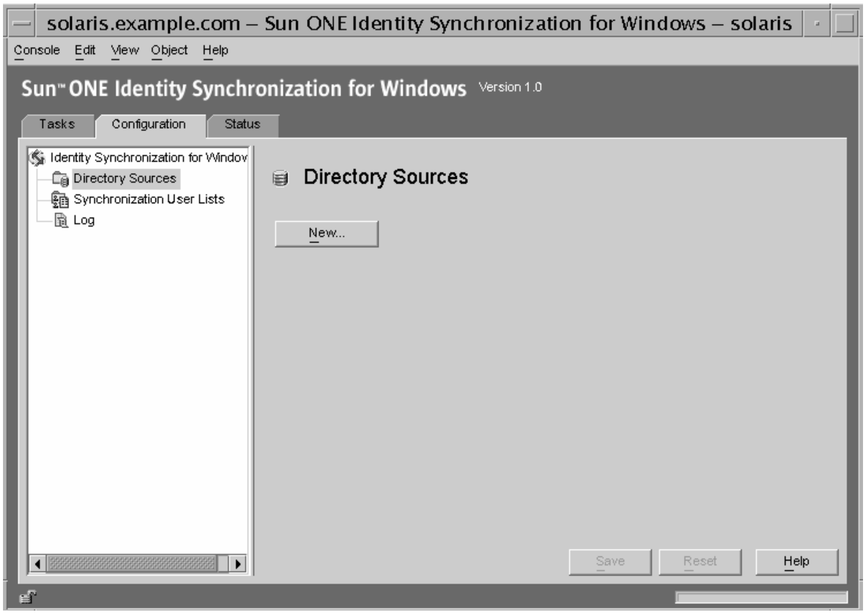
Creating Directory Sources

1. Select the Directory Sources node in the navigation tree.

   The Directory Sources window appears. Create the directory sources in this order (if applicable):

   - Sun ONE Directory Source
   - Active Directory Source
   - NT SAM Directory Source

NOTE At least one Sun ONE Directory source and at least one Active Directory source or NT SAM directory source is the minimum required configuration.


solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris
Console Edit View Object Help
Sun™ ONE Identity Synchronization for Windows Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
Synchronization User Lists
Log
Directory Sources
New...
Save
Reset
Help

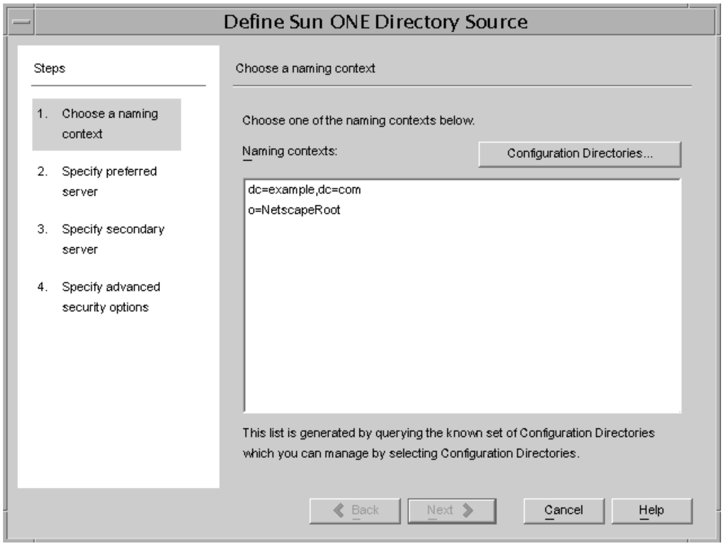
Sun ONE Directory Source

1. If you would like to create a new Sun ONE directory source press the top New button and select Sun Directory source from the drop-down list.

2. This invokes the directory sources creation wizard.

   This queries the list of configuration directory sources to find out what other directories exist and displays them. You may also select Configuration Directories to in order to specify additional references to configuration directories in your enterprise. This wizard queries the specified configuration directories in the discovery of naming contexts and directory servers in an enterprise.

NOTE Only one user database is supported per Sun ONE Directory Server source.



Define Sun ONE Directory Source

Steps

1. Choose a naming context
2. Specify preferred server
3. Specify secondary server
4. Specify advanced security options

Choose a naming context

Choose one of the naming contexts below.

Naming contexts:

dc=example,dc=com
o=NetscapeRoot

This list is generated by querying the known set of Configuration Directories which you can manage by selecting Configuration Directories.

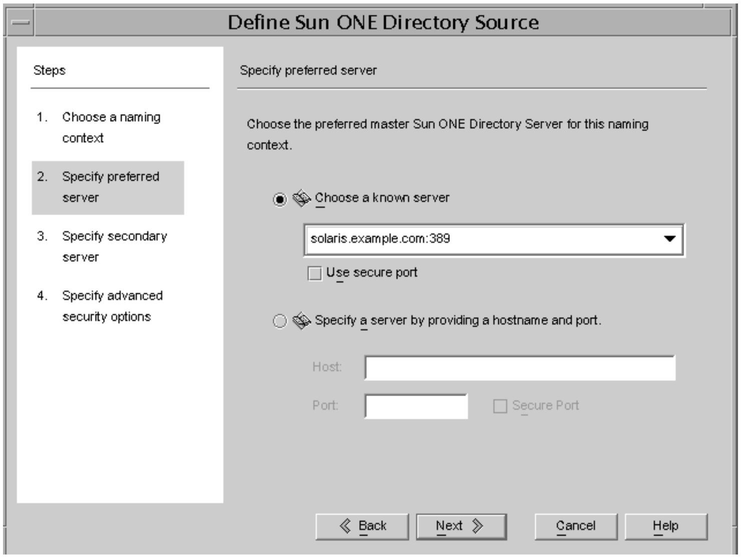
3. Select the naming context (fully qualified distinguished name) of the desired Sun directory source and press Next.

   If you do not see the naming context where identity entries which you are interested in synchronizing identities exist, select Configuration Directories as noted in the previous step.

   Sun ONE Directory Server creates a naming context whose prefix corresponds to the components of the machine's DNS domain entry. It uses the following suffix:

   `dc=`*`your_machine's_DNS_domain_name`*`.`

   That is, if your machine domain is example.com, then you should configure the suffix **`dc=example, dc=com`** for your server. The entry named by the chosen suffix must already exist in the directory.



Define Sun ONE Directory Source

Steps

1. Choose a naming context
2. Specify preferred server
3. Specify secondary server
4. Specify advanced security options

Specify preferred server

Choose the preferred master Sun ONE Directory Server for this naming context.

● Choose a known server

solaris.example.com:389

□ Use secure port

○ Specify a server by providing a hostname and port.

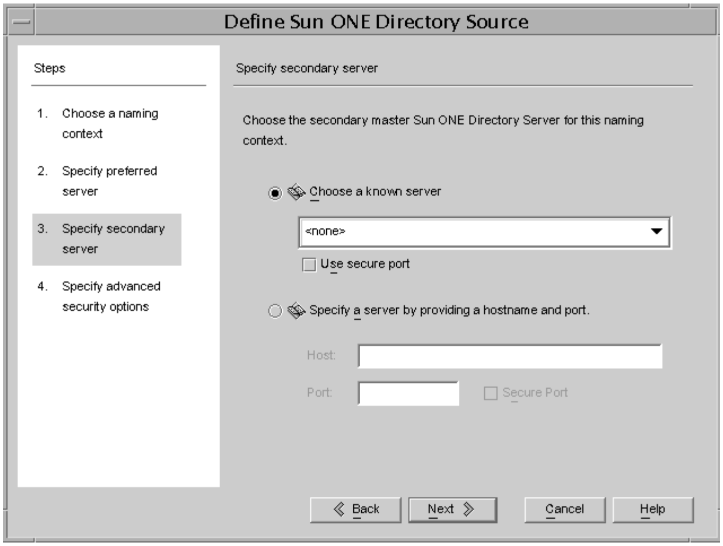
Host:

Port: □ Secure Port

4. Designate a preferred Directory Server from the drop-down list or specify the desired preferred server by hostname and port. Press Next.

NOTE If a Directory Server is not running, it will not appear in the drop-down list of available hosts. If the host in question is down temporarily, specify the necessary information in the Specify a server by providing a hostname and port field.

Check the Use Secure Port box if you want secure SSL communication. If SSL is enabled, then there are additional setup requirements. See Configuring Security for more information. Press Next.



Define Sun ONE Directory Source

Steps

1. Choose a naming context
2. Specify preferred server
3. Specify secondary server
4. Specify advanced security options

Specify secondary server

Choose the secondary master Sun ONE Directory Server for this naming context.

● Choose a known server

<none>

□ Use secure port

○ Specify a server by providing a hostname and port.

Host:

Port: □ Secure Port

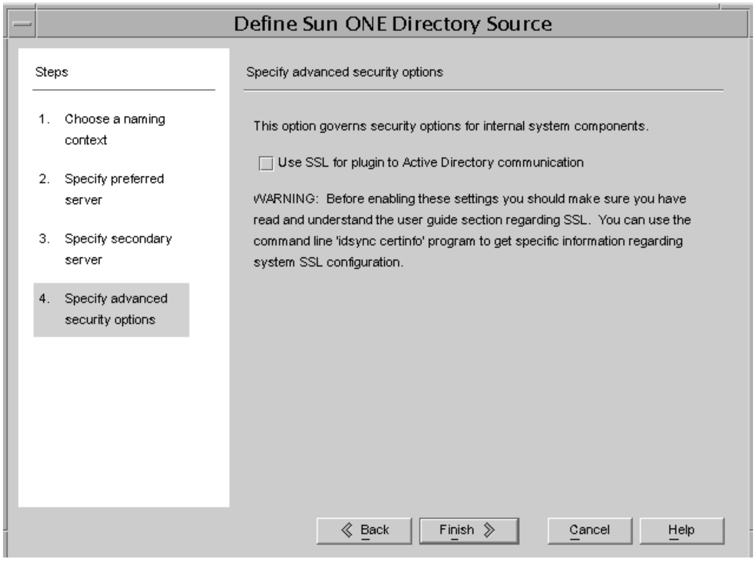
5. If desired, designate a secondary Directory Server from the drop-down list or specify the desired master server by hostname and port. Press Next.

NOTE	You should not use the same hostname and port for both the preferred and the secondary server in a Sun Directory Source.

NOTE	If the Directory Server is not running, it will *not* appear in the drop down list of available hosts. If the host in question is down temporarily, specify the necessary information in the Specify a server by providing a hostname and port field.

6. Specify any advanced security options. Check the appropriate box if you want secure SSL communication for plugin to Active Directory communication.

NOTE	If SSL is enabled, then there are additional setup requirements. See Configuring Security for more information. If your primary and secondary Directory Servers are part of a Multi-Master Replication (MMR) deployment, refer to Installation Notes for Replicated Environments for more instructions.



Define Sun ONE Directory Source

Steps

1. Choose a naming context
2. Specify preferred server
3. Specify secondary server
4. Specify advanced security options

Specify advanced security options

This option governs security options for internal system components.

□ Use SSL for plugin to Active Directory communication

WARNING: Before enabling these settings you should make sure you have read and understand the user guide section regarding SSL. You can use the command line 'idsync certinfo' program to get specific information regarding system SSL configuration.

7. Press Finish.

   Your selection is added to the topology tree under Directory Sources and a summary window of configuration information appears. Review your choices, and, if necessary, press Edit Servers to make any changes.

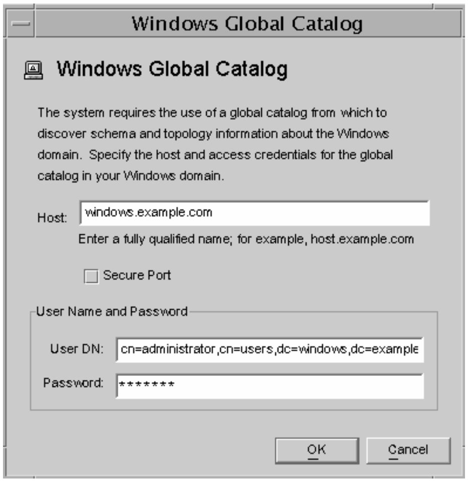
8. Repeat these procedures to add all Sun ONE directory sources in your network.

Active Directory Source

Perform the following steps if there are Windows Active Directory servers in you network:

1. Select the Directory Sources node in the navigation tree.
2. Press the top New button and select Active Directory source from the drop-down list.

   The Windows Global Catalog window appears if a Global catalog has not been already configured.



Windows Global Catalog

Windows Global Catalog

The system requires the use of a global catalog from which to discover schema and topology information about the Windows domain. Specify the host and access credentials for the global catalog in your Windows domain.

Host: windows.example.com

Enter a fully qualified name; for example, host.example.com

□ Secure Port

User Name and Password

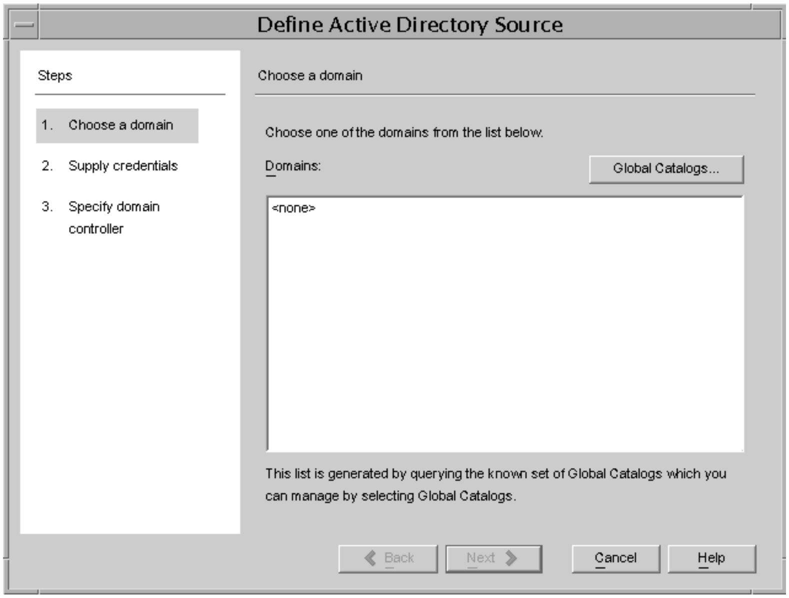
User DN: cn=administrator,cn=users,dc=windows,dc=example

Password: *******

3. Enter the Host name of the system containing the Windows Global Catalog.
4. Enter the complete User distinguished name of the administrator; for example:

   `cn=administrator,cn=users,dc=example,dc=com`
5. Enter the administrator password and press OK.
6. The Define Active Directory Source wizard appears.

   This wizard queries the Active Directory global catalog to find out what other domains exist and displays them. You may also select Global Catalogs to specify or create a different domain and credentials.



Define Active Directory Source

Steps

1. Choose a domain
2. Supply credentials
3. Specify domain controller

Choose a domain

Choose one of the domains from the list below.

Domains:

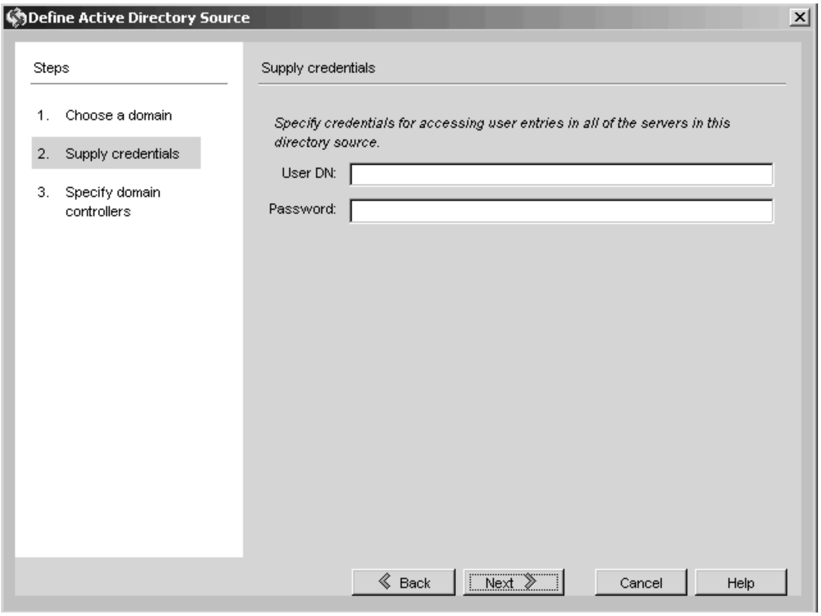
<none>

This list is generated by querying the known set of Global Catalogs which you can manage by selecting Global Catalogs.

7. Select the desired domain controller and press Next.

NOTE You cannot use the same hostname and port for an Active Directory Source's domain controller and for a preferred or secondary server in a Sun Directory Source.

NOTE If the selected Active Directory domain has multiple domain controllers, then the domain controller with the Primary Domain Controller FSMO role should be selected for synchronization. By default, password changes made at all domain controllers are replicated immediately to the Primary Domain Controller FSMO role owner, and if this domain controller is selected, Identity Synchronization for Windows will synchronize these password changes immediately to the Sun ONE Directory Server. This feature can be disabled by setting the `AvoidPdcOnWan` attribute in the Windows registry, but this will significantly delay password synchronization. See Microsoft Knowledge Base Article 232690 for more information.



Specify credentials for accessing user entries in all of the servers in this directory source.

User DN:

Password:

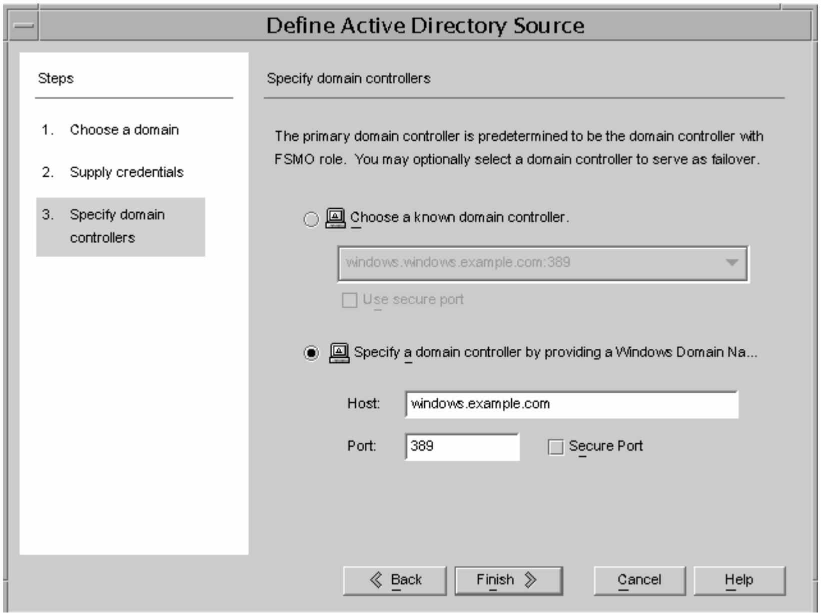
8. Enter the complete User distinguished name of the administrator; for example:

```
cn=administrator,cn=users,dc=example,dc=com
```

9. Enter the administrator password and press Next.

When configuring an Active Directory source, the administrator must provide a user name and password for the Active Directory user that the connector will use to connect to Active Directory. The user's minimum rights will depend on the direction of synchronization, as follows:

- If you are configuring synchronization flow from Active Directory to Directory Server only, then the user provided for the Active Directory connector does not require many special privileges. A normal user with the extra privilege to "Read All Properties" in the domain being synchronized will suffice.
- If you are configuring synchronization flow from Directory Server to Active Directory, then the connector user must have more privileges because synchronization changes user entries in Active Directory. In this setup, the connector user must have either the "Full Control" privilege or be a member of the Administrators group.



Define Active Directory Source

Steps

1. Choose a domain
2. Supply credentials
3. Specify domain controllers

Specify domain controllers

The primary domain controller is predetermined to be the domain controller with FSMO role. You may optionally select a domain controller to serve as failover.

○ Choose a known domain controller.

windows.windows.example.com:389

□ Use secure port

● Specify a domain controller by providing a Windows Domain Na...

Host: windows.example.com

Port: 389 □ Secure Port

10. Select the desired domain controller from the drop-down list or press the radio button, Specify a domain controller by providing a Window Domain name and port. Specify a host and port number.

11. Check the Use secure port box if you want secure SSL communication. Press Finish.

NOTE The installer automatically installs the CA certificate in the Active Directory connector if you are using Microsoft certificate server. If you are not then you must manually add the CA certificate in the Active Directory connector (see Enabling SSL in the Active Directory Connector). If you change your flow settings after initial configuration these procedures apply as well.

Your selection is added to the navigation tree under Directory Sources and a summary window of configuration information appears. Review your choices, and if necessary, press Edit Controller to make changes.

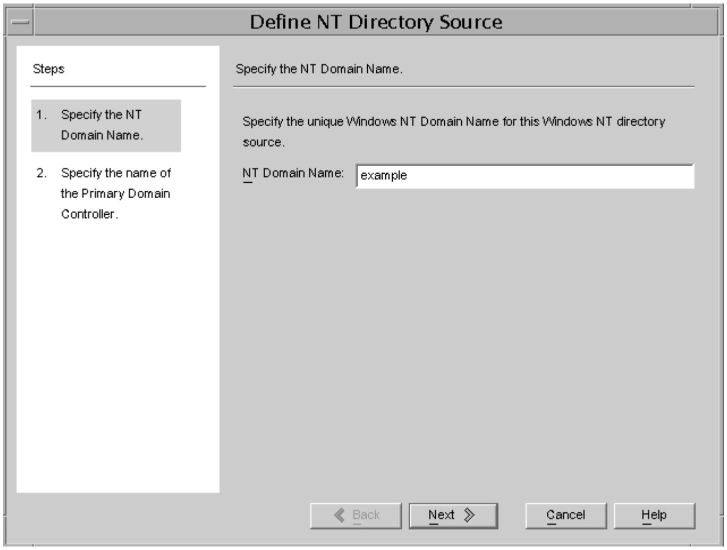
12. Add an Active Directory source for each Windows domain in your network.

NT SAM Directory Source

If deploying Identity Synchronization for Windows on NT platforms designate the NT SAM Directory source as follows:

1. Select the Directory Sources node in the navigation tree.

2. Press the top New button and select NT SAM Directory source from the drop-down list. The Define NT Directory Source wizard appears.



Define NT Directory Source

Steps

1. Specify the NT Domain Name.

2. Specify the name of the Primary Domain Controller.

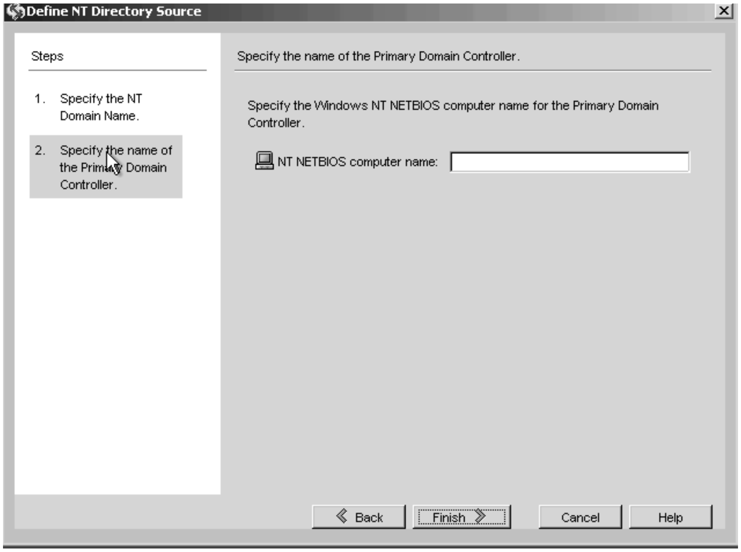
Specify the NT Domain Name.

Specify the unique Windows NT Domain Name for this Windows NT directory source.

NT Domain Name: example

3. Enter the unique NT domain name for the NT directory source. Press Next.

   This information can be determined from your NT host by right-clicking Network Neighborhood>Properties>Identification.



Define NT Directory Source

Steps

1. Specify the NT Domain Name.
2. Specify the name of the Primary Domain Controller.

Specify the name of the Primary Domain Controller.

Specify the Windows NT NETBIOS computer name for the Primary Domain Controller.

NT NETBIOS computer name:

4. Enter the NT NETBIOS computer name for the preferred NT directory source. Press Finish.

   This information can be determined from your NT host by right-clicking Network Neighborhood>Properties>Identification.

   Your selection is added to the navigation tree under Directory Sources and a configuration window appears. Review your choices, press Edit to make changes.

5. Add an NT directory source for each Windows NT machine in your network.

Deleting Directory Sources

If you must delete a directory source, use the following steps:

1. Before you can delete the directory source, you must first delete all of the Synchronized User Lists associated with that source.

   a. Right-click on the Synchronization User List listed under the Synchronization User List node in the topology tree.

b. When the pop-up menu displays, select Delete to remove the SUL.

NOTE You can preserve the information in the SUL by associating the SUL with a *different* directory source, as described on page 84.

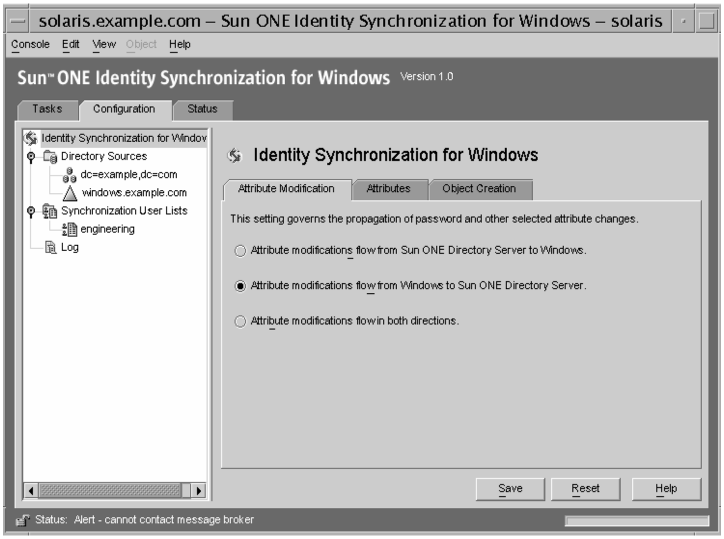
---

2. Right-click on the directory source name listed under the Directory Sources node in the topology tree.

3. When the pop-up menu displays, select Delete to remove the directory source.

Setting Attribute Modification Flow

1. Select Identity Synchronization for Windows at the highest level in the topology tree. Press the Attribute Modification tab.


solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris
Console
Edit
View
Object
Help
Sun™ ONE Identity Synchronization for Windows
Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
dc=example,dc=com
windows.example.com
Synchronization User Lists
engineering
Log
Identity Synchronization for Windows
Attribute Modification
Attributes
Object Creation
Save
Reset
Help

This setting governs the propagation of password and other selected attribute changes.

○ Attribute modifications flow from Sun ONE Directory Server to Windows.

● Attribute modifications flow from Windows to Sun ONE Directory Server.

○ Attribute modifications flow in both directions.

Status: Alert - cannot contact message broker

There are three choices for how password and other specified user attributes flow between systems changes flow between systems:

- Attribute Modifications flow from the Sun ONE Directory Server to Windows. Select this if you wish user password changes made in the Sun ONE Directory Server environment to propagate to Windows servers.

- Attribute Modifications flow from Windows to Sun ONE Directory Server. (Default) Select this if you wish user password changes made in the Windows environment to propagate to Sun ONE Directory Servers.
- Attribute Modifications flow in both directions. Select this if you wish user password changes made in either environment to propagate to the other.

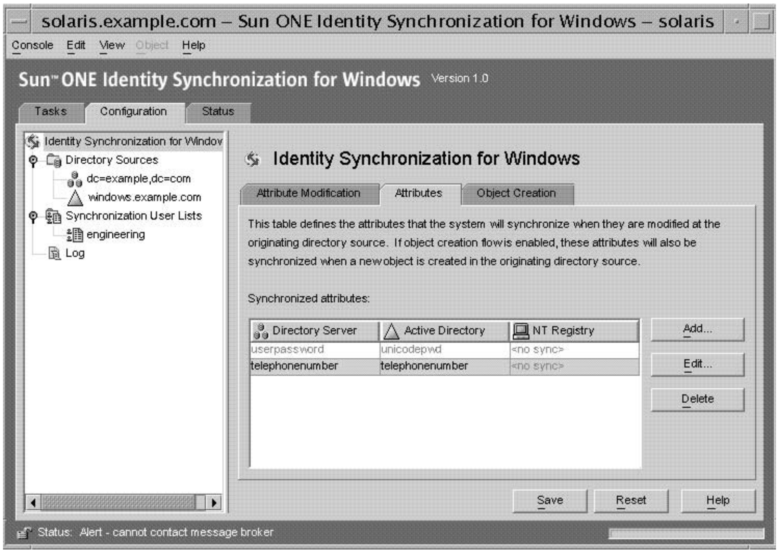
2. Select the desired modification flow.

Setting the Modification Attribute Mapping

Based on the chosen object class, a series of attributes are available for both Directory Server and Active Directory. These chosen attributes will be synchronized. Using procedures in this section, select the desired properties under Active Directory and an equivalent attribute will display under Directory Server.

The user selects one-to-one mappings of the user entry attributes that ought to be synchronized between the Sun ONE Directory Server and the Windows Active Directory and NT environments. There is a default list of attributes provided (minimum set required to sync password).

1. Select Identity Synchronization for Windows at the top of the topology tree.
2. Press the Attributes tab.


solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris
Console
Edit
View
Object
Help
Sun™ ONE Identity Synchronization for Windows
Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
dc=example,dc=com
windows.example.com
Synchronization User Lists
engineering
Log
Identity Synchronization for Windows
Attribute Modification
Attributes
Object Creation
Add...
Edit...
Delete
Save
Reset
Help

This table defines the attributes that the system will synchronize when they are modified at the originating directory source. If object creation flow is enabled, these attributes will also be synchronized when a new object is created in the originating directory source.

Synchronized attributes:

Directory Server	Active Directory	NT Registry
userpassword	unicodepwd	<no sync>
telephonenumber	telephonenumber	<no sync>

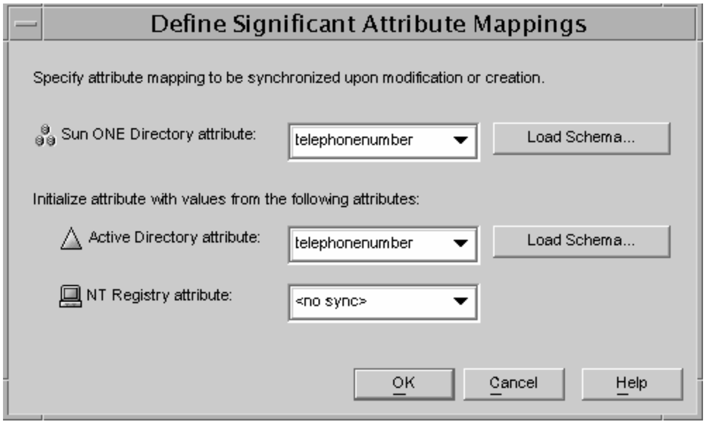
Status: Alert - cannot contact message broker

3. Press Add.

   The Sun ONE User Object Class dialog box appears.

4. From the drop-down menu, choose the desired user object class that will be used in the Sun ONE scheme. Click OK.

   The Define Significant Attribute Mapping window appears.



Define Significant Attribute Mappings

Specify attribute mapping to be synchronized upon modification or creation.

Sun ONE Directory attribute: telephonenumber

Initialize attribute with values from the following attributes:

Active Directory attribute: telephonenumber

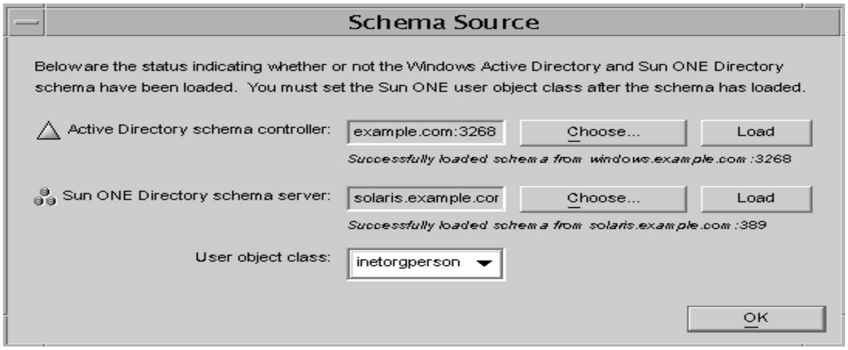
NT Registry attribute: <no sync>

NOTE Note that the drop-down lists show <no sync> and are inactive. The attribute lists are disabled until the schema has been loaded.

5. Press Load Schema.

   The Schema Source window appears.

NOTE The Sun schema is loaded from the configuration directory by default but can be loaded from an alternative directory source. The Active Directory schema is loaded as soon as the first Global Catalog has been specified in the system. If no Global Catalog has been specified to this point, you must specify one as directed in the next step.



Beloware the status indicating whether or not the Windows Active Directory and Sun ONE Directory schema have been loaded. You must set the Sun ONE user object class after the schema has loaded.

Active Directory schema controller: example.com:3268

Successfully loaded schema from windows.example.com:3268

Sun ONE Directory schema server: solaris.example.cor

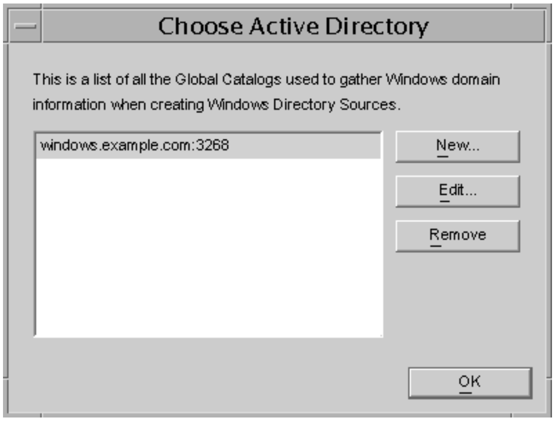
Successfully loaded schema from solaris.example.com:389

User object class: inetorgperson

6. If necessary, press the Choose button next to the missing directory schema controller.

   If you press the choose button for the Active Directory schema controller, the Choose Active Directory Schema Host window appears.

   If you press the choose button for the sun ONE schema controller, the Choose Sun ONE Directory Schema Host window appears. Step 7 through Step 9 are the same for both Active Directory and Sun ONE Schema.



This is a list of all the Global Catalogs used to gather Windows domain information when creating Windows Directory Sources.

windows.example.com:3268

7. Select the desired Active Directory instance and press OK.

8. At Schema Source window select the desired User object class from the drop-down list. Press OK.

9. At the Define Significant Attributes Mappings window select the desired value from the Sun ONE Directory attribute drop-down list.

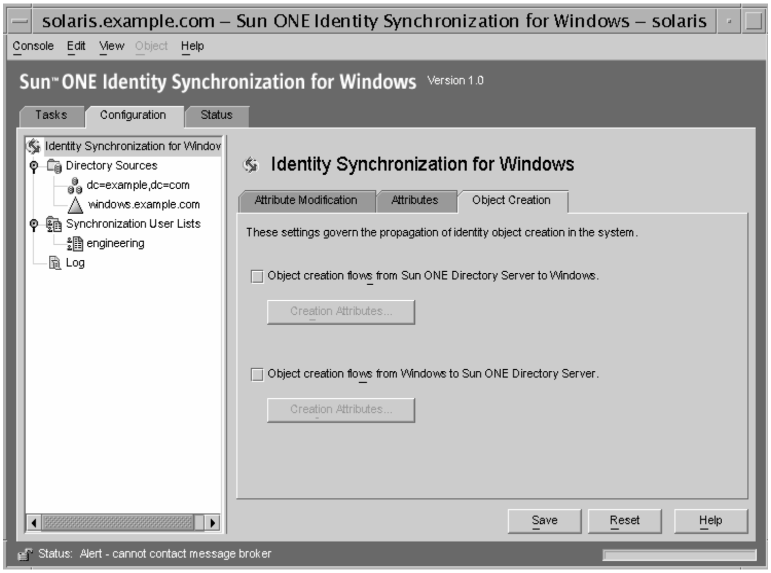
10. Select the desired value from the Active Directory or NT SAM Registry attribute drop-down list.

11. Press OK.

12. To designate additional attributes, repeat Step 9 through Step 11 for specifying a mapping between the Sun One Directory Server and Active Directory or NT Registry.

Setting Object Creation Flow and Attribute Mapping

1. Press the Object Creation tab.



These settings govern the propagation of identity object creation in the system.

□ Object creation flows from Sun ONE Directory Server to Windows.

□ Object creation flows from Windows to Sun ONE Directory Server.

Status: Alert - cannot contact message broker

There are two choices for how newly created users can propagate between systems:

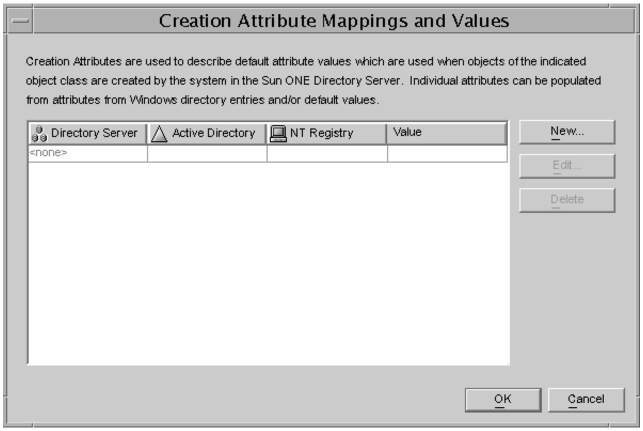
- **Object creation flows from Sun ONE Directory Server to Windows.** Select this if you wish users created in the Sun ONE Directory Server environment and have them propagate to Windows Active Directory servers.
- **Object creation flows from Windows to Sun ONE Directory Server.** Select this if you wish users created in the Windows Active Directory environment and have them propagate to the Sun ONE Directory Server.

2. Select the desired creation flow.

3. Press the Creation Attributes under the selected creation flow.

   The Creation Attribute Mappings and Values window appears. The following procedure is for object creation flows from Windows to Sun ONE Directory Server. Creation flow from Sun ONE Directory Server to Windows is done in a similar manner.

NOTE In order to satisfy schema constraints regarding required attributes for user object classes, it may be necessary to specify additional attributes to flow through the system during a user creation. Note that this is not necessary if the required attributes have been specified as modification attributes as described in the section Setting the Modification Attribute Mapping.



Creation Attribute Mappings and Values

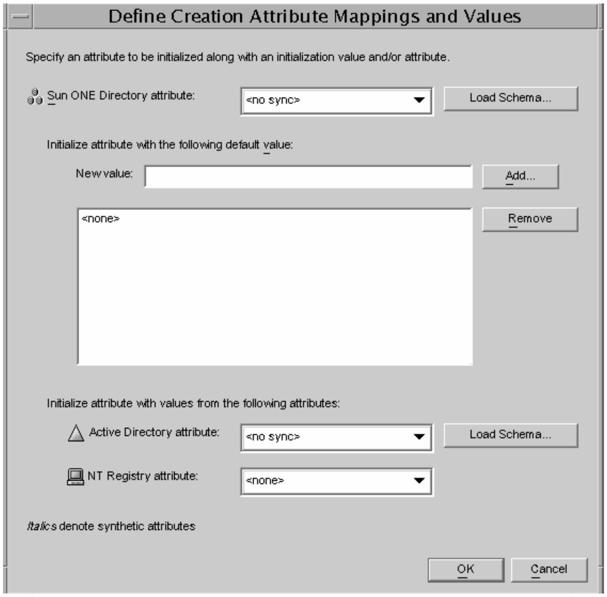
Creation Attributes are used to describe default attribute values which are used when objects of the indicated object class are created by the system in the Sun ONE Directory Server. Individual attributes can be populated from attributes from Windows directory entries and/or default values.

Directory Server	Active Directory	NT Registry	Value
<none>			

Note that one of the drop-down lists shows <no sync> and is inactive.

4. Press New.

   The Define Creation Attribute Mappings and Values window appears.



Define Creation Attribute Mappings and Values

Specify an attribute to be initialized along with an initialization value and/or attribute.

Sun ONE Directory attribute: <no sync>

Initialize attribute with the following default value:

New value:

<none>

Initialize attribute with values from the following attributes:

Active Directory attribute: <no sync>

NT Registry attribute: <none>

Italics denote synthetic attributes

5. Select the desired value from the Sun ONE Directory attribute drop-down list.

6. If you want to initialize with a default value, enter that value in New value and press add. It is added to the value listing field.

   To remove a value from the listing select the desired value and press Remove.

7. If necessary, select the desired value from the Active Directory or NT SAM Registry attribute drop-down list.

8. Press OK.

   Repeat Step 4 through Step 8 to select cn and sn at this time.

9. To designate additional attributes, repeat Step 5 through Step 8 for specifying a mapping between the Sun One Directory Server and Active Directory or NT.

Creating Synchronization User Lists

A Synchronization User List (SUL) specifies the domain of users in two directory sources to be synchronized. This section contains the following:

- Overview
- Defining Synchronization User Lists

Overview

Every Synchronized User List contains two definitions that identify which users in a directory to synchronize, which users to exclude from synchronization, and where to create new users. One definition identifies which Sun ONE Directory Server users to synchronize and the other identifies the Windows users to synchronize.

NOTE To synchronize users in a Sun ONE Directory Server with multiple Active Directory domains, you must define one SUL for each Active Directory domain.

For more information about SULs, including components of a definition, how to define multiple SULs, how multiple SULs are processed, and how to configure multiple Windows domain support refer to Appendix D, "Synchronization User List Definitions and Configuration" on page 213.

The following procedure uses the console to identify and link user types between servers.

NOTE When there are existing users, you must run the `idsync resync` script after running `idsync linkusers`. If you do not resynchronize existing users, the resynchronization behavior remains undefined.

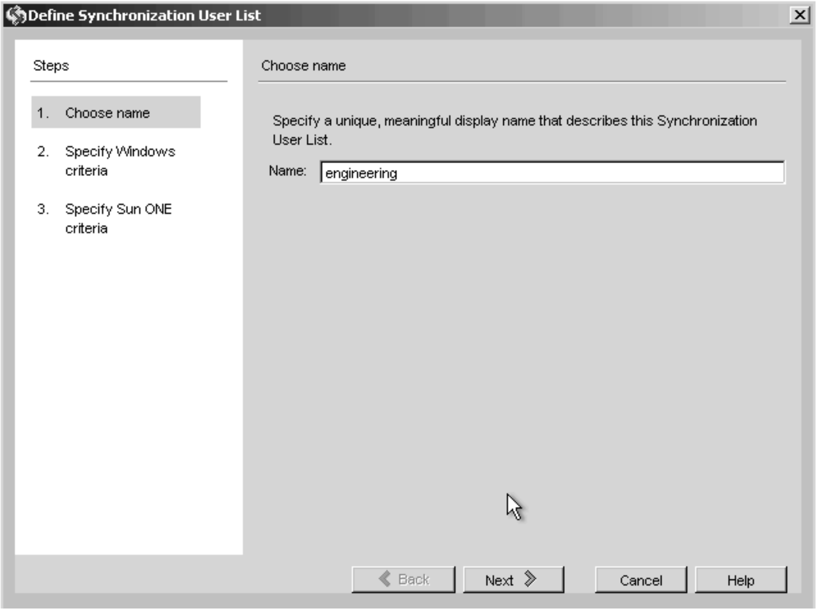
For more information about the `idsync resync` script, see "User Resynchronization" on page 111.

For additional information on linking existing users after connectors have been installed using the `idsync linkusers` script see "Linking Users" on page 106 of this document.

Defining Synchronization User Lists

1. Select the Synchronization User Lists node in the topology tree.
2. Press New Synchronization User List.

   The Define Synchronization User List wizard appears.

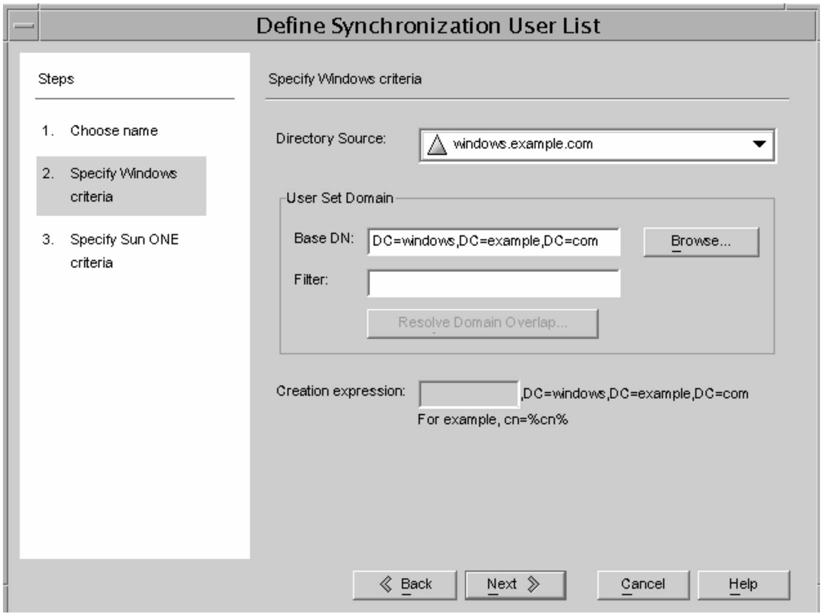


Define Synchronization User List
Steps
1. Choose name
2. Specify Windows criteria
3. Specify Sun ONE criteria
Choose name
Back
Next
Cancel
Help

Specify a unique, meaningful display name that describes this Synchronization User List.

Name: engineering

3. Enter an appropriate name for the new list and press Next.



Define Synchronization User List

Steps

1. Choose name
2. Specify Windows criteria
3. Specify Sun ONE criteria

Specify Windows criteria

Directory Source: windows.example.com

User Set Domain

Base DN: DC=windows,DC=example,DC=com

Filter:

Creation expression: ,DC=windows,DC=example,DC=com

For example, cn=%cn%

4. Select a Windows Directory Source from the drop-down list.

5. Enter the User Set Domain's Base DN setting, either by typing into the text field directly or by pressing the Browse button, which invokes a directory browser. For example:

   `DC=example,DC=com`

NOTE No base DN or creation expression is allowed for NT machines.

6. If desired, enter a Filter to specify which users under this base DN are synchronized or not.

   If you have the same base DN for multiple synchronization user lists, you may want to use a filter to distinguish between them. The filter follows LDAP query syntax except that it allows only exact matches; for example `(&(o=sales)(st=CA))` could be used to select users in the California sales organization.

7. If necessary, press Resolve Domain Overlap if users exist on multiple domains.

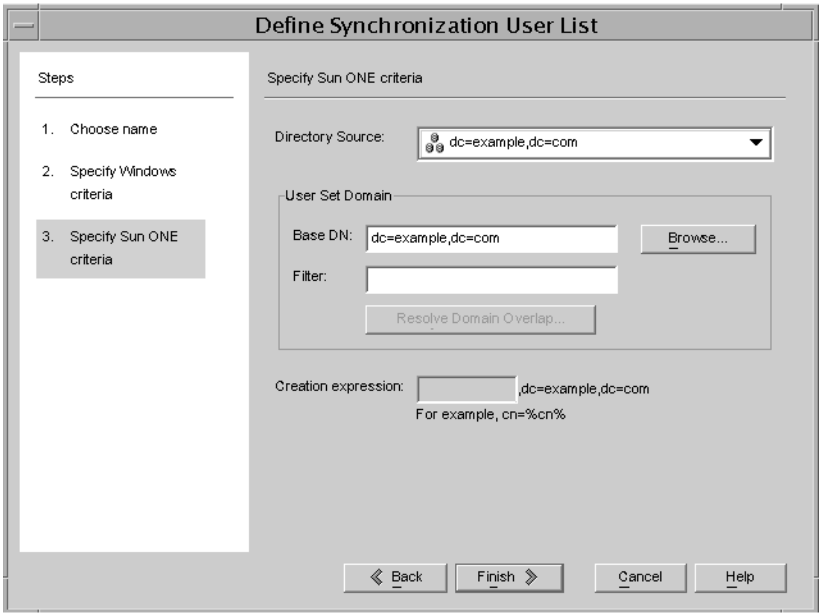
   Use Resolve Domain Overlap to define a preference for the synchronization user list in case a user matches multiple lists. (For more information, see Understanding Synchronized User List Definitions.)

8. If allowed enter a creation expression for all Windows Active Directory synchronized user lists; for example:

   `cn=%cn%,cn=hostname ou=Domain Controllers,dc=example,dc=com`

NOTE A creation express defines the parent DN and naming attribute used when new entries are propagated from Active to Sun Directory. A Sun creation expression is only allowed if object creation has been configured to flow from Active to Sun Directory (See Setting Object Creation Flow and Attribute Mapping).

9. Press Next.



Define Synchronization User List

Steps
1. Choose name
2. Specify Windows criteria
3. Specify Sun ONE criteria

Specify Sun ONE criteria

Directory Source: dc=example,dc=com

User Set Domain

Base DN: dc=example,dc=com

Filter:

Creation expression: ,dc=example,dc=com

For example, cn=%cn%

10. Select a Sun ONE Directory Source from the drop-down list.

11. Enter the User Set Domain's Base DN setting, either by typing into the text field directly or by pressing the Browse button, which invokes a directory browser. For example:

    `CN=hostname,OU=Domain Controllers,DC=example,DC=com`

12. If desired, enter a Filter to specify which users under this base DN are synchronized or not.

13. Press Resolve Domain Overlap if users exist on multiple domains.

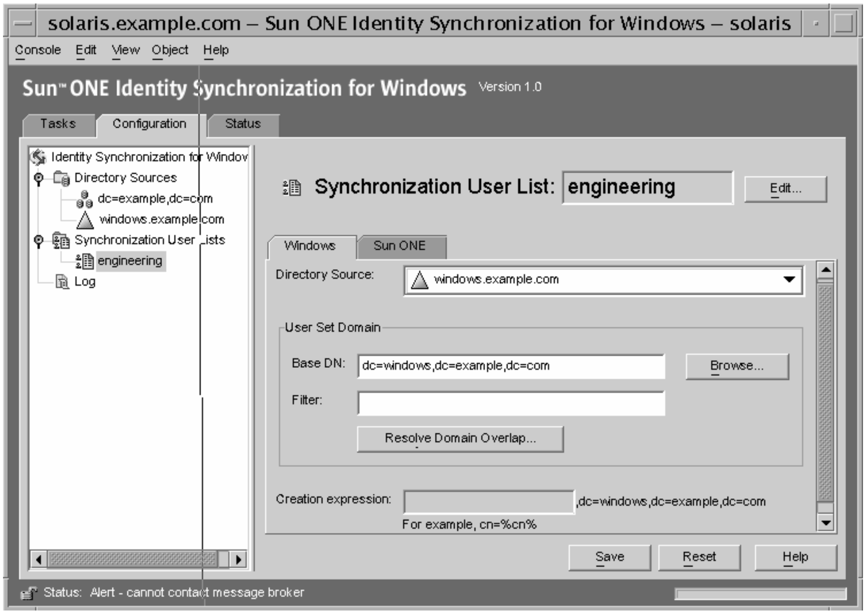
14. If allowed enter a creation expression for all Sun ONE Directory Server synchronized user lists; for example:

```
uid=%uid%,ou=people,dc=example,dc=com
```

NOTE A creation express defines the parent DN and naming attribute used when new entries are propagated from Active to Sun Directory. A Sun creation expression is only allowed if object creation has been configured to flow from Active to Sun Directory (See Setting Object Creation Flow and Attribute Mapping).

15. Press Finish.

16. The new synchronization user list is added to the navigation tree and the Configuration > Synchronization User List menu appears.



solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris

Sun™ ONE Identity Synchronization for Windows Version 1.0

Synchronization User List: engineering

Directory Source: windows.example.com

Base DN: dc=windows,dc=example,dc=com

Filter:

Creation expression: ,dc=windows,dc=example,dc=com

For example, cn=%cn%

Status: Alert - cannot contact message broker

17. Create a Synchronization User List that includes every directory source in your network except for the Directory Server.

NOTE Before you can delete a directory source, you must first delete all of the Synchronized User Lists associated with that source. However, you can preserve the information in a Synchronized User List by associating the SUL with a *different* Directory Source, as follows:

- To associate an SUL with a new directory source:

  a. Create the new directory source.

  b. Edit the SUL and replace the existing directory source with the new directory source.

  c. Delete the old directory source.

- To associate an SUL with an existing directory source:

  a. Edit the SUL and replace the existing directory source with the desired directory source.

  b. Delete the old directory source.

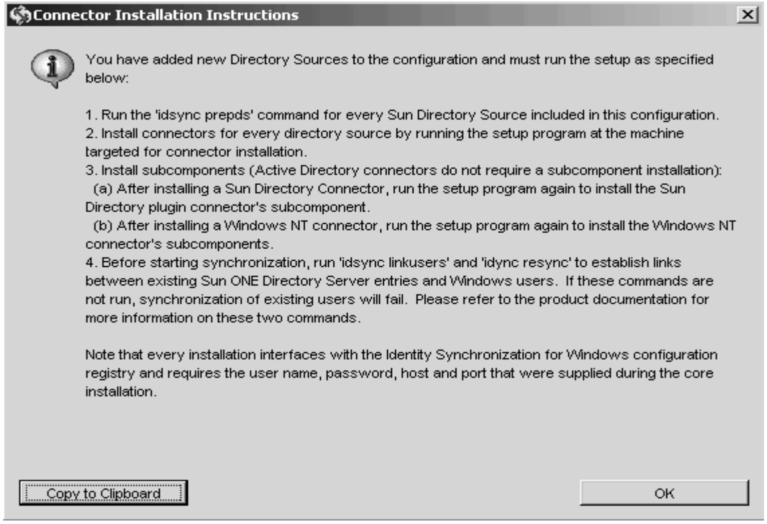
Saving a Configuration

1. Press Save to store your settings at this point.

2. The Configuration Validity Status window appears stating that the configuration is valid. Press continue to save the configuration.

NOTE Configuration validation is checked before being saved. Configuration validation errors appear in red, while warnings appear in yellow. The configuration cannot be saved with errors. You should attempt to clear warnings but configurations can be saved with them present.

3. A dialog box appears giving instructions on how to proceed in installing connectors and subcomponents. Read carefully and press OK.



Connector Installation Instructions

You have added new Directory Sources to the configuration and must run the setup as specified below:

1. Run the 'idsync prepds' command for every Sun Directory Source included in this configuration.
2. Install connectors for every directory source by running the setup program at the machine targeted for connector installation.
3. Install subcomponents (Active Directory connectors do not require a subcomponent installation):
   (a) After installing a Sun Directory Connector, run the setup program again to install the Sun Directory plugin connector's subcomponent.
   (b) After installing a Windows NT connector, run the setup program again to install the Windows NT connector's subcomponents.
4. Before starting synchronization, run 'idsync linkusers' and 'idync resync' to establish links between existing Sun ONE Directory Server entries and Windows users. If these commands are not run, synchronization of existing users will fail. Please refer to the product documentation for more information on these two commands.

Note that every installation interfaces with the Identity Synchronization for Windows configuration registry and requires the user name, password, host and port that were supplied during the core installation.

Prepare Directory Server

`idsync prepds` is a command-line program that prepares a Sun ONE Directory Server source for use by Identity Synchronization for Windows. Run the `prepds` program after planning for a Identity Synchronization for Windows configuration because `prepds` requires the administrator to know which hosts and suffixes will be used.

Change directory to:

```
cd /sunone/servers/isw-hostname/bin
```

The following is a command-line example:

```
/sunone/servers/isw-hostname/bin/idsync prepds -h hostname -p 33827
-D “cn=Directory Manager” -w password -s dc=example,dc=sun,dc=com
```

Table 4-1 lists the `idsync prepds` arguments for preparing the Sun ONE Directory Server and their definitions.

Table 4-1 `idsync prepds` Arguments

Argument	Description
-h	This is the DNS name of the Directory Server server-instance serving as the preferred host.
-p	This is the port of the DS server-instance serving as the preferred host.
-j	This is the DNS name of the DS server-instance serving as the secondary host.
-r	This is the port of the DS server-instance serving as the secondary host.
-D	This is DN of the Directory Manager user. (Required)
-w <– or password>	This is the password for the Directory Manager user or ‘-‘ which indicates the password should be read from standard input (Required). Use the ‘-‘ argument so that the password does not appear on the command line. You can place the password in a file and protect that file appropriately. You can then pipe that file in the command so that the password doesn’t appear on the terminal.

Argument	Description
-s	This is the name of the rootsuffix to use for adding the index. (Required) A rootsuffix is a distinguished name such as `dc=example,dc=com`. A single synchronized user database is supported on each host even in a multi-database Directory Server deployment. Ensure that users to be synchronized are within one database.
-x	If this not provided, the program will add the equality and presence indices for the dspswuserlink attribute to the synchronized database of each specified host. If it is provided, the indexes will not be added. Note that the administrator must create the indices before installing the Directory Server Connector. This option is provided because index creation places the synchronized database in read-only mode and may take long time if the database contains large number of entries. Use this option to use other means of creating indices. Take the Directory Server off-line and add the index using LDAP tools. This could save some time, depending on the deployment. Refer to the *Sun ONE Directory Server Administrator's Guide* for information about creating indexes using the Directory Server Console. Create an index of the "Equality and Presence" type called `dspswuserlink` in the objectclass `DSPSWUser`.
-Z	Specify that SSL be used to provide certificate-based client authentication.
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Netscape[tm] Communicator, if available; for example: `-P /home/uid/.netscape/cert7.db` When using the command on the same host as the directory server, you may use the server's own certificate database, for example: `-PinstallDir/slapd-serverID/alias/cert7.db`. Use the `-P` option alone to specify server authentication only.
-m <secmod db path>	Specify the path to the security module database. For example: /var/Sun/mps/slapd-serverID/secmodule.db You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Accessing the Directory Server Via SSL

`idsync prepds` lists the following options that provide information about securely accessing the Directory Server via SSL:

```
[-Z] [-P <cert db path>] [-m <secmod db path>]
```

idsync prepds Results

Upon successful execution you should see the following message:

```
The preferred host is running a supported version of Sun ONE
Directory Server.
Application is using the Directory Manager credentials at the
preferred host.
The Retro Changelog database is available at the preferred host.
The required schema elements are already present.
The Connector user has been created on the preferred host.
The Retro Changelog access control instance is already present on
the preferred host.
The Connector user access control instance is already present on the
preferred host.
The equality index is already present on the preferred host.
SUCCESS: Sun Directory Source is ready for synchronization. Please
install the Directory Server Connector.

SUCCESS
```

If this message is not seen re-execute the script.

Continuing the Installation

Upon completion of initial core configuration, use the setup program to install Identity Synchronization for Windows connectors following procedures found in the Chapter 5, “Connector and Subcomponent Installation”.

Chapter 5

Connector and Subcomponent Installation

This section contains Identity Synchronization for Windows connector and subcomponent installation procedures including:

- Directory Server Connector and Plugin Subcomponent
- Windows Active Directory Connector
- Windows NT Connector and Subcomponent

Install connectors by running the setup program on the chosen machine.

The following connectors require a follow-up subcomponent installation:

- Directory Server connector
- Windows NT connector

After installing a Directory Server connector, run the setup program again to install the Directory Server plugin subcomponent. The Directory Server plugin must be installed in each Directory Server master and consumer that stores users being synchronized. After installing a Windows NT connector, run setup again on that machine to install the NT subcomponent. Active Directory connectors do not have subcomponents.

NOTE For best performance, make sure the Java 2 Platform Standard Edition SDK (JDK) 1.4.1_03 or higher and not the Java Runtime Environment (JRE) is available on the host on which connector installation is being performed.

Directory Server Connector and Plugin Subcomponent

Use the setup program to install Directory Server connectors. After installing a Directory Server connector, run the setup program again to install the Directory Server plugin subcomponent.

Connectors can be installed in the same system as core or another system. the plugin must be installed on the system where Directory Server is installed.

Directory Server Connectors

1. Run the setup program on the desired Directory Server. Find it in the directory `installer` where you untarred binary files. On Windows machines execute the setup executable in the installer directory:

   ```
   cd installer
   setup.exe
   ```

   Execute the installer on UNIX machines:

   ```
   cd installer
   ./runInstaller.sh
   ```

2. At the Welcome screen press Next.

3. At the Software License Agreement screen read the license and press Yes (Accept License) to accept the license terms. Press No to exit setup.

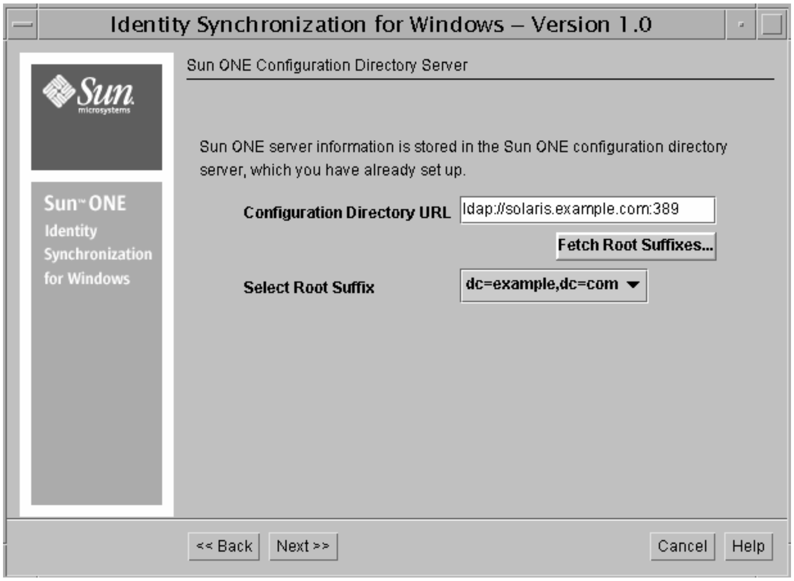
4. When prompted, enter the Configuration Directory URL.

   The configuration directory is the Directory Server instance where Identity Synchronization for Windows stores the core configuration information. Enter the following:

   `ldap:`*`//Directory Server name`*`:`*`port number`*

5. Select the root suffix for the configuration directory. Press Fetch Root Suffixes and a drop-down list will populate with choices. Select the desired root suffix.

   The root suffix that is chosen is the root suffix where the configuration is stored, which may be different than the rootsuffix being synchronized.



Sun ONE Configuration Directory Server

Sun ONE server information is stored in the Sun ONE configuration directory server, which you have already set up.

Configuration Directory URL: ldap://solaris.example.com:389

Select Root Suffix: dc=example,dc=com

6. Press Next.

7. Enter the Configuration Directory Server administrator name and password. Press Next.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

8. Enter the configuration password. Press Next.

   If installing the connector on the same Solaris or Windows system as the core a message appears stating that setup has detected that core or connectors have already been installed on the system. All additional components will be installed under `c:\Program Files\Sun\MPS\isw-`*`hostname`*`\`. Press OK.

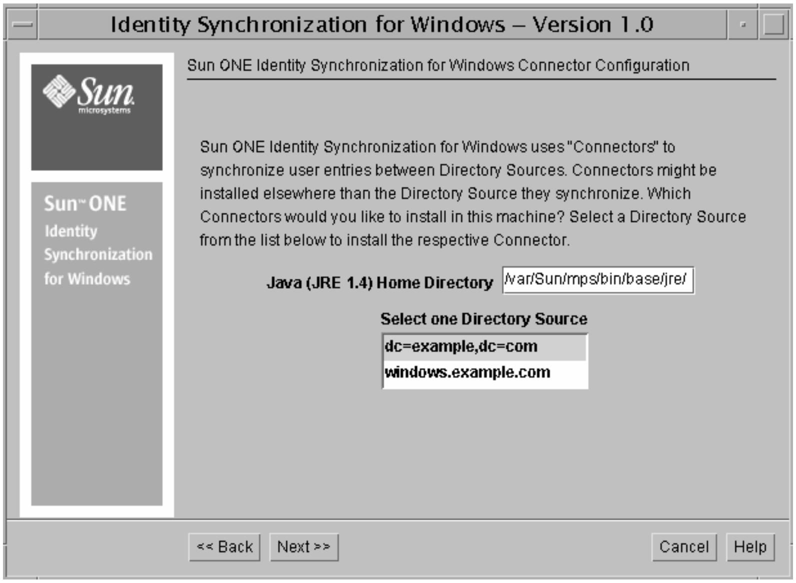
9. Enter the Java home directory. Note that at a minimum this directory must contain a Java 1.4.1_03 installation and should be the Java 2 Platform Standard Edition SDK (JDK) and not the Java Runtime Environment (JRE) for best performance.

   For example on Windows platforms:

```
C:\j2sdk1.4.1_04
```

On UNIX platforms:

```
/usr/j2sdk1.4.1_04/j2se
```



Identity Synchronization for Windows – Version 1.0
Sun microsystems
Sun™ ONE Identity Synchronization for Windows
<< Back
Next >>
Cancel
Help

Sun ONE Identity Synchronization for Windows Connector Configuration

Sun ONE Identity Synchronization for Windows uses "Connectors" to synchronize user entries between Directory Sources. Connectors might be installed elsewhere than the Directory Source they synchronize. Which Connectors would you like to install in this machine? Select a Directory Source from the list below to install the respective Connector.

Java (JRE 1.4) Home Directory: /var/Sun/mps/bin/base/jre/

Select one Directory Source:
- dc=example,dc=com
- windows.example.com

10. Select the appropriate directory sources from the drop-down list.

    Identity Synchronization for Windows uses connectors to synchronize user passwords between directory sources. Example directory sources are:

Directory Source	Example Entry
Sun ONE Directory Server	`dc=example,dc=com`
Windows Active Directory	`example.com`
Windows NT	`EXAMPLE`

    Choose the directory source from a Sun ONE Directory Server with which you wish to connect. Press Next.

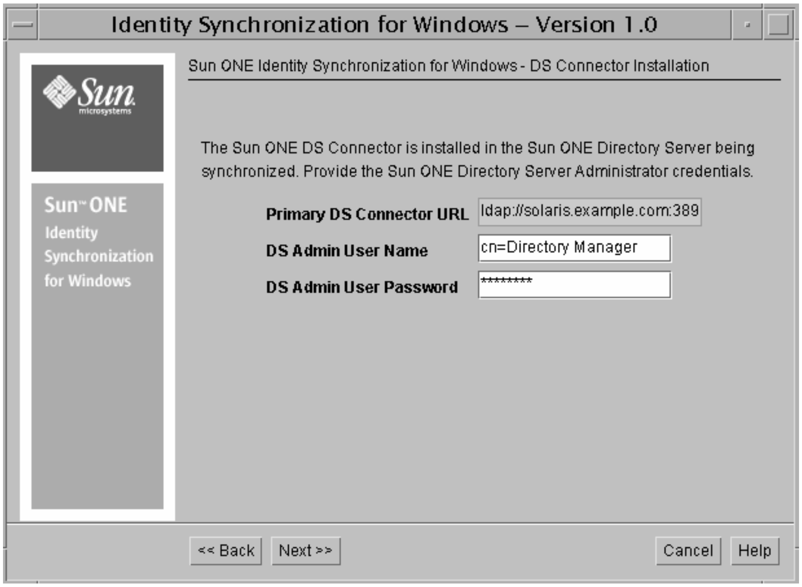
    The Directory Server Connector Installation window appears.

11. Select the Primary Directory Server Connector URL shown in the window.

    This URL contains the LDAP port for the Directory Server being synchronized.

12. Enter the Directory Server administrator name and password. Press Next.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.



Identity Synchronization for Windows – Version 1.0

Sun ONE Identity Synchronization for Windows - DS Connector Installation

The Sun ONE DS Connector is installed in the Sun ONE Directory Server being synchronized. Provide the Sun ONE Directory Server Administrator credentials.

Primary DS Connector URL ldap://solaris.example.com:389

DS Admin User Name cn=Directory Manager

DS Admin User Password *********

The Directory Port configuration window appears.

13. Enter the fully qualified Localhost Name with domain.

14. Enter the Port Number of the connector.

Choose an available server port, which the connector will use to securely pass configuration information to the Directory Server plugin.Press next.

NOTE Use a non-SSL, non-secure LDAP port number for the Directory Server(s) while installing connectors.

---

15. The setup program checks for available disk space and an installation summary menu appears.

Ensure that the following component appears in the summary menu.

`DSConnector`

16. When ready, click Install Now.

    An installation status bar and the Register Configuration Data window appears.

17. When prompted, press Next to register with the selected Directory Server. This may take several minutes.

18. An Installation Summary appears. Press Details if you wish to view the installation log. Press Close to exit setup.

19. Perform Step 1 through Step 18 for each additional Directory Server connector.

Directory Server Subcomponent

1. Run the setup program again from each machine where a directory server is installed.

NOTE	Only install a subcomponent from the machine which has the Directory Source for which the plugin is intended.

2. At the Welcome screen press Next.
3. At the Software License Agreement screen read the license and press Yes (Accept License) to accept terms of license. Press No to exit setup.
4. When prompted enter the Configuration Directory URL.

   The configuration directory is the Directory Server instance where Identity Synchronization for Windows configuration information is stored. Enter the following:

   `ldap://`*`Directory Server name`*`:`*`port number`*

5. Select the root suffix for the configuration directory. Press Fetch Root Suffixes and a drop-down list will populate with choices. Select the desired root suffix.

   The root suffix that is chosen is the root suffix where the configuration is stored, which may be different than the rootsuffix being synchronized.

6. Press Next.
7. Enter the Configuration Directory Server administrator name (`cn=Directory Manager`) and password. Press Next.

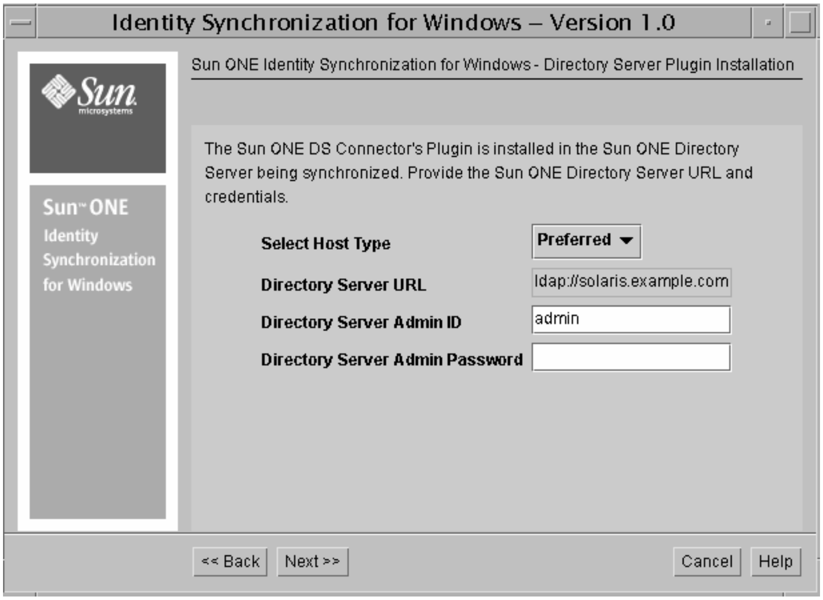
NOTE	The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

8. Enter the configuration password. Press Next

   If installing the subcomponent or connector on the same system as the core a message appears stating that setup has detected that core or connectors have already been installed on the system. All additional components will be installed under the installation directory. Press OK.

9. Select Subcomponents and press Next.

10. Select the directory source for this subcomponent installation. Press Next.



Identity Synchronization for Windows – Version 1.0

Sun ONE Identity Synchronization for Windows - Directory Server Plugin Installation

The Sun ONE DS Connector's Plugin is installed in the Sun ONE Directory Server being synchronized. Provide the Sun ONE Directory Server URL and credentials.

Select Host Type: Preferred

Directory Server URL: ldap://solaris.example.com

Directory Server Admin ID: admin

Directory Server Admin Password:

11. Select the appropriate Host Type from the drop-down menu.

- Preferred
- Secondary
- Other

NOTE All Directory Server replicas (those other than preferred / secondary servers) also need a Directory Server subcomponent installed. Choose Other as the Host Type for these Directory Server replicas.

12. Enter the Directory Server administrator's name and password. Press Next.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

13. The setup program checks for available disk space and an installation summary screen appears.

    Ensure that the following subcomponent appears in the summary menu.

    ```
    DSSubcomponents
    ```

14. When ready, press Install Now.

    An installation status bar and the Register Configuration Data window appears.

15. When prompted, press Next to register with the selected Directory Server. This may take several minutes.

    A window appears stating that you must restart the Directory Server where the plugin has been installed. Press OK.

16. An Installation Summary appears. Press Details if you wish to view the installation log. Press close to exit setup.

17. Restart the Directory Server where the plugin has been installed.

Windows Active Directory Connector

NOTE Verify network functionality before proceeding. Specifically, determine whether servers in your network can communicate with the Configuration Directory Server.

1. Download Identity Synchronization for Windows to each system on which you wish to install a connector. Refer to “Starting the Installer” on page 49.

2. Run the setup program on the desired server. Find it in the directory where you untarred binary files. On Windows machines, execute the setup executable in the installer directory:

    ```
    cd installer
    setup.exe
    ```

    Execute the installer on UNIX machines:

    ```
    cd installer
    ./runInstaller.sh
    ```

3. At the Welcome screen press Next.
4. At the Software License Agreement screen read the license and press Yes (Accept License) to accept terms of license. Press No to exit setup.
5. When prompted, enter the Configuration Directory URL.

   The configuration directory is the Directory Server instance where Identity Synchronization for Windows configuration information is stored. Enter the following:

   `ldap://`*`Directory Server name`*`:`*`port number`*

   Press Fetch Root Suffixes and a drop-down list will populate with choices. Select the desired root suffix. Press Next.

   The root suffix that is chosen is the root suffix where the configuration is stored, which may be different than the rootsuffix being synchronized.
6. Enter the Configuration Directory Server administrator name and password. Press Next.

NOTE	The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

7. Enter the configuration password. Press Next.

   If installing the connector on the same Windows system as core, a message appears stating that setup has detected that core or connectors have already been installed on the system. All additional components will be installed under the installation directory. Press OK.
8. Select Connector and press Next.
9. Enter the Java home directory. Note that at a minimum this directory must contain a Java 1.4.1_03 installation and should be the Java 2 Platform Standard Edition SDK (JDK) and not the Java Runtime Environment (JRE) for best performance.
10. Select the appropriate directory source from the list.

    Identity Synchronization for Windows uses connectors to synchronize user passwords between directory sources. Example directory sources are:

Table 5-1 Directory Source Examples

Directory Source	Example Entry
Sun ONE Directory Server	`dc=example,dc=com`
Windows Active Directory	`example.com`

Choose the directory source from a Windows Active Directory with which you wish to connect.

11. Press Next.

12. If the connector is configured to use LDAP over SSL to communicate with Active Directory, enter the Administrator's password.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

NOTE The default prompt for the user is Administrator. You can use the cn of any user under cn=users in Active Directory (for example `cn=<user>,cn=users,dc=domain,dc=com`).

However, the user value has to exist in the cn=users container in Active Directory.

---

13. Press Get Certificate Authorities. Select the Certificate Authority from the drop-down menu. Press Next

14. The setup program checks for available disk space and an installation summary menu appears.

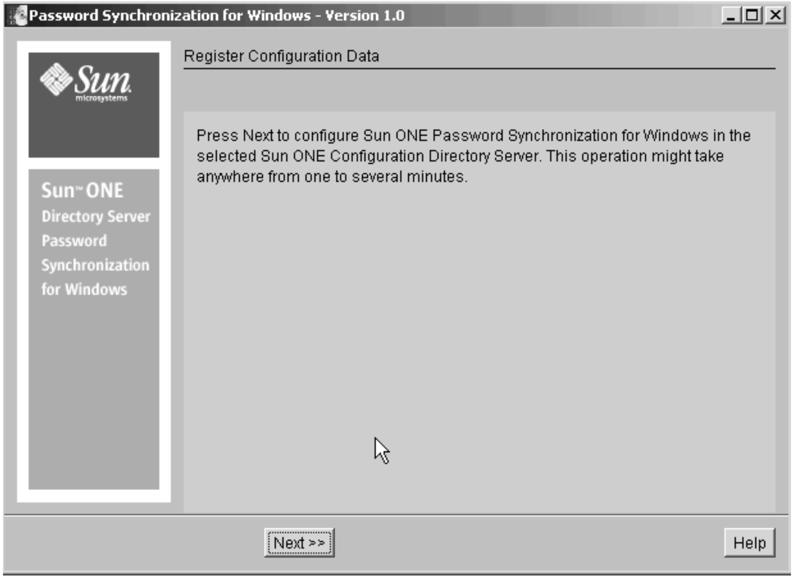
    Ensure that the following component appears in the summary menu.

    `ADConnector`

15. When ready press Install Now.

    An installation status bar and the Register Configuration Data window appears.

16. When prompted, press next to register with the selected Directory Server. This may take several minutes.



Register Configuration Data

Press Next to configure Sun ONE Password Synchronization for Windows in the selected Sun ONE Configuration Directory Server. This operation might take anywhere from one to several minutes.

17. An Installation Summary appears. Press Details if you wish to view the installation log. Press Close to exit setup.

18. Perform Step 1 through Step 17 for each additional Active Directory domain where you wish to install a connector.

Windows NT Connector and Subcomponent

Use the setup program to install Windows NT connectors. After installing a Windows NT connector, run the setup program again to install the Windows NT subcomponents.

NOTE The Windows NT connector and subcomponents must be installed on the machine where the Windows NT directory source is installed. The Windows NT directory source must be on a machine that is a primary domain controller.

Windows NT Connector

1. Download Identity Synchronization for Windows to the primary domain controller of each Windows NT domain to be synchronized. Refer to “Starting the Installer” on page 49.

2. Run the setup program on the desired server. Find it in the directory where you untarred binary files. Execute the setup executable in the installer directory:

   ```
   cd installer
   ```

   ```
   setup.exe
   ```

3. At the Welcome screen press Next.

4. At the Software License Agreement screen read the license and press Yes (Accept License) to accept terms of license. Press No to exit setup.

5. When prompted enter the Configuration Directory URL.

   The configuration directory is the Directory Server instance where Identity Synchronization for Windows configuration information is to be stored. Enter the following:

   `ldap:`*`//Directory Server name`*`:`*`port number`*

   Press Fetch Root Suffixes and a drop-down list will populate with choices. select the desired root suffix. Press Next. The root suffix that is chosen is the root suffix where the configuration is stored, which may be different than the rootsuffix being synchronized.

6. Enter the Configuration Directory Server administrator name and password. Press Next.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

7. Enter the configuration password. Press Next.

8. Enter the directory in which to install the connector. Press Next.

   A message appears if the directory does not exist. Press Yes to create a new directory.

9. Select Connectors and press Next.

10. Enter the Java home directory. Note that at a minimum this directory must contain a Java 1.4 installation and should be the Java 2 Platform Standard Edition SDK (JDK) and not the Java Runtime Environment (JRE) for best performance.

11. Select the appropriate directory source from the drop-down list.

    Identity Synchronization for Windows uses connectors to synchronize user passwords between directory sources. Example directory sources are:

Table 5-2 Directory Source Examples

Directory Source	**Example Entry**
Windows NT	`EXAMPLE`
Sun ONE Directory Server	`dc=example,dc=sun,dc=com`

    Choose the directory source from a Windows NT directory with which you wish to connect.

12. Press Next.

13. Enter the fully qualified Localhost Name.

14. Enter the Port Number of the connector. Press next.

    Choose an available server port, which the connector will use to securely pass configuration information to the Windows NT subcomponent.

15. The setup program checks for available disk space and an installation summary menu appears.

    Ensure that the following component appears in the summary menu.

    `NTConnector`

16. When ready, press Install Now.

    An installation status bar and the Register Configuration Data window appears.

17. When prompted, press next to register with the selected Directory Server. This may take several minutes.

18. An Installation Summary appears. Click Details if you wish to view the installation log. Press Close to exit setup.

Windows NT Subcomponent

1. Run the setup program again from the primary domain controller.
2. At the Welcome screen press Next.
3. At the Software License Agreement screen read the license and press Yes (Accept License) to accept the license terms. Press No to exit setup.
4. When prompted enter the Configuration Directory URL.

   The configuration directory is the Directory Server instance where Identity Synchronization for Windows configuration information is stored. Enter the following:

   `ldap://Directory Server name:port number`

5. Select the root suffix for the configuration directory. Press Fetch Root Suffixes and a drop-down list will populate with choices. Select the desired root suffix.

   The root suffix that is chosen is the root suffix where the configuration is stored, which may be different than the rootsuffix being synchronized.

6. Press Next.
7. Enter the Configuration Directory Server administrator name and password. Press Next.

NOTE The credentials provided will be sent without encryption. Consider changing them in the Directory Server after installation if network traffic confidentiality may be compromised.

8. Enter the configuration password. Press Next.

   A message appears stating that setup has detected that core or connectors have already been installed on the system. All additional components will be installed under the installation directory. Press OK.

9. If a window appears that gives you a choice between connectors and subcomponents, select Subcomponents and press Next.
10. Select the directory source for this subcomponent installation. Press Next.
11. The setup program checks for available disk space and an installation summary menu appears.

    Ensure that the following component appears in the summary.

`NTSubcomponents`

12. When ready, click Install Now.

    An installation status bar and the Register Configuration Data window appears.

13. When prompted, press Next to register with the selected Directory Server. This may take several minutes.

    A message appears stating that the Windows subcomponents have been installed. Please reboot the machine. Click Next.

14. An Installation Summary appears. Click Details if you wish to view the installation log. Press Close to exit setup.

15. Reboot the NT Server where the subcomponent has been installed.

16. Perform the procedures in "Windows NT Connector" and "Windows NT Subcomponent" for each additional NT domain where you wish to install a connector.

NOTE For the Windows NT SAM Change Detector subcomponent to be effective, you must turn on the NT audit log. Under Start > Programs > Administrative Tools > User Manager, select Policies > Audit Policies.
Select Audit These Events and then both the Success and Failure boxes for User and Group Management.

Under Event Log Settings in the Event Viewer>Event Log Wrapping, select Overwrite Events as Needed.

Chapter 6

Synchronizing Existing Users

This chapter contains information for linking and resynchronizing existing users for new Identity Synchronization for Windows installations. Users should be linked after core and connector installation has been completed.

This chapter includes the following sections:

- "Linking Users" on page 106
- "User Resynchronization" on page 111
- "Starting and Stopping Synchronization" on page 115
- "Starting and Stopping Services" on page 116

Identity Synchronization for Windows provides two command line utilities that bootstrap deployments with existing users:

idsync linkusers. This command uses administrator-specified matching rules to pair up existing entries

idsync resync. This command can populate an empty directory with the contents of a remote directory or bulk synchronize attribute values between two existing user populations.

This table summarizes the post-installation steps to follow based on existing user populations:

Table 6-1 Dealing With Existing Users

Users Exist In		**Post-installation Steps to Follow**	
Windows	**Sun ONE Directory Server**	**Existing users should be synchronized**	**Existing users should NOT be synchronized**
No	No	None	None

Users Exist In		Post-installation Steps to Follow	
Windows	Sun ONE Directory Server	Existing users should be synchronized	Existing users should NOT be synchronized
No	Yes	Run `idsync resync -o Sun -c` to create existing Directory Server users in Windows	None
Yes	No	Run `idsync resync -c` to create existing Windows users in Directory Server	Run `idsync resync -u` to populate the connector's local cache of user entries.
Yes	Yes	Run `idsync linkusers` to link users between Windows and Directory Server. Then run `idsync resync` or `idsync resync -o` Sun to synchronize existing user values between the two directories.	Run `idsync resync -u` to populate the connector's local cache of user entries.

Linking Users

Prior to starting synchronization for your network ensure that all existing users are linked between directory sources. The `idsync linkusers` command enables administrators to link existing users in two directory sources. Run this command after all connectors have been installed. The directory administrator provides rules for matching users between the two directories (for example: for a user entry to be linked in the two directories both the first names and last names must match in both directory entries).

NOTE When there are existing users, you must run the `idsync resync` command after running `idsync linkusers`. If you do not resynchronize existing users, the resynchronization behavior remains undefined.

For more information about the `idsync resync` command, see "User Resynchronization" on page 111.

This section describes how to run this command in the following sections:

- Usage
- idsync linkusers Central Log
- idsync linkusers Caveats

Once Active Directory and Directory Server have been populated with users and the Active Directory and Directory Server connectors have been installed (but synchronization has not been started), the `idsync linkusers` command can be used to link the existing users.

Usage

The `idsync linkusers` command accepts the following arguments:

Table 6-2 `idsync linkusers` Usage

Argument	**Description**
-h	Configuration Directory hostname
-p	Configuration Directory port number
-D	Directory Manager bind distinguished name
-w	Directory Manager Bind password
-s	This is the name of the root suffix to use for adding the index. (Required) A root suffix is a distinguished name such as `dc=example,dc=com`. A single synchronized user database is supported on each host even in a multi-database Directory Server deployment. Ensure that users to be synchronized are within one database.
-q	Configuration password
-f	XML filename for `linkusers`
[-a <ldap-filter>]	Specify an LDAP filter to control the linking of individuals or groups.
-n	Run in safe mode. This will only log what would have happened if the `linkusers` operation were run. This allows you to preview the effects of an `linkusers` operation.
-Z	Specify that SSL be used to provide certificate-based client authentication.

Argument	Description
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

To link the existing users using the sample `IlodeLinkUsersIntegrate.cfg` file (see LinkUsers XML Document Sample for a listing of this file),

Linking user entries and resolving data conflicts could be described as more art than science. There are many reasons why the `idsync linkusers` command might fail to link two users in opposing directory sources and depends to a large extent on the consistency of the data in the linked directories.

One strategy for using `idsync linkusers` is to use the "safe mode" flag to refine the linking criteria gradually until an optimum set of user matching criteria is found (see Appendix B) for a reference in defining matching criteria). You should realize however that there is a balance to be achieved through linkage accuracy and linkage coverage. For instance, if both directory sources contain an employee ID or social security number, you should probably begin with linking criteria that includes this number only. You might feel that in order to improve linkage accuracy, you might include a last name attribute in the criteria as well. However, you may then lose linkages for entries that would have matched on ID alone due to inconsistent last name values in the data. These objects will have to be manually linked. To do manual linking of entries could be done in any number of ways involving your favorite LDAP tools including browsers and command-line tools.

In the following example, if you have specified a user matching criteria specifying linkages to be established between user object having the same value for the sn and givenname attributes. Proceed as follows:

1. run `idsync linkusers` as follows:

   ```
   root@example:/var/sun/mps/isw-hostname/bin $ idsync linkusers -h
   hostname -p 389 -D "cn=Directory Manager" -w dirmanager -s
   dc=example,dc=com -q password -f
   ../samples/ILodeLinkUsersIntegrate
   User linking operation started. Enter ’c’ to cancel.
   Progress: Dumped=7, Previously linked=0, Successfully linked=2,
   Failed to link=4
   Success
   ```

2. View the linking log file at under the logs directory:

   ```
   [2002/12/11 21:47:01.267 -0600] INFO 14 CNN101 davide
   "Destination - Matched Remote Entry=’[sn=Smith,
   employeenumber=1000, givenname=Alice] [uid=AliceSmith]’to Local
   Entry=’[Action operation =9, Action DN = CN=Alice
   Smith,OU=people,DC=central,DC=example,DC=com, attrname =
   givenname, attrValue = Alice(hex value: 416C696365), op type = 0,
   attrname = sn, attrValue = Smith(hex value: 536D697468), op type
   = 0, attrname = uid, attrValue = AliceSmith(hex value:
   416C696365536D697468), op type = 0, Attribute name = nsuniqueid,
   Attribute value = c2c0a184-1dd111b2-80a5faf5-9800ff15(hex value:
   63326330613138342D31646431313162322D38306135666166352D3938303066
   663135)

   [2002/12/11 21:47:01.708 -0600] INFO 14 CNN101 davide
   "Destination Failed to Link Entry=’[sn=Hayes,
   employeenumber=1002, givenname=Charlie] [uid=Charlie Hayes]’"
   (Action ID=6, SN=6)

   [2002/12/11 21:48:28.182 -0600] INFO 21 CNN101 davide
   "Destination - Entry=’[sn=Smith, employeenumber=1000,
   givenname=Alice] [uid=AliceSmith]’ already linked." (Action
   ID=13, SN=4)
   ```

3. Notice in the above example that Charlie Hayes entry did not match. The first thing to suspect in such a case would be data consistency. You might try locating the user entry in the directory using `ldapsearch` and searching for entries where sn=Hayes:

```
root@example:/var/sun/mps/isw-hostname/bin $ ldapsearch -D
"cn=directory manager" -w dirmanager -h hostname -b
dc=central,dc=example,dc=com sn=Hayes
```

```
uid=CHayes,ou=People, dc=central,dc=example,dc=com
uid=CHayes
objectClass=top
objectClass=person
objectClass=organizationalPerson
objectClass=inetorgperson
sn=Hayes
cn=Charles Hayes
givenName=Charles
```

4. This search reveals the linking problem to be one of data inconsistency. The simplest way to resolve this problem would be to change the givenname attribute value from Charles to Charlie or visa versa on the opposite directory source. Continue trying to resolving data inconsistencies for all match failures or refine the user match criteria and rerun `idsync linkusers` until you have achieved an acceptable percentage of entry linkages.

idsync linkusers Central Log

The result of all `idsync linkusers` operations are reported in a special central log named linking.log. This log lists all of the users that were properly linked, those that failed to link, and those that were previously linked. An example of each is shown below. The "already linked" case was generated by rerunning the same `idsync linkusers` operation.

```
[2002/12/11 21:47:01.267 -0600] INFO    14  CNN101 davide  "Destination - Matched
Remote Entry='[sn=Smith, employeenumber=1000, givenname=Alice] [uid=AliceSmith]'
to Local Entry='[Action operation =9, Action DN = CN=Alice
Smith,OU=people,DC=central,DC=sun,DC=com, attrname = givenname, attrValue =
Alice(hex value: 416C696365), op type = 0, attrname = sn, attrValue = Smith(hex
value: 536D697468), op type = 0, attrname = uid, attrValue = AliceSmith(hex value:
416C696365536D697468), op type = 0, Attribute name = nsuniqueid, Attribute value =
c2c0a184-1dd111b2-80a5faf5-9800ff15(hex value:
63326330613138342D31646431313162322D38306135666166352D39383030666663135)
```

```
[2002/12/11 21:47:01.708 -0600] INFO    14  CNN101 davide  "Destination Failed to
Link Entry='[sn=Hayes, employeenumber=1002, givenname=Charlie] [uid=Charlie
Hayes]'" (Action ID=6, SN=6)
```

```
[2002/12/11 21:48:28.182 -0600] INFO    21  CNN101 davide  "Destination -
Entry='[sn=Smith, employeenumber=1000, givenname=Alice] [uid=AliceSmith]' already
linked." (Action ID=13, SN=4)
```

Note some pre-existing special Active Directory users such as Administrator and Guest might appear in this log as failures.

idsync linkusers Caveats

Beware of the following when running `idsync linkusers`:

Indexed Attributes

Attributes used in an `idsync linkusers` operation should be indexed. If there are multiple attributes in a UserMatchingCriteria and at least one of them is indexed, then performance will probably be acceptable. If no attributes in a UserMatchingCriteria are indexed, then performance will be unacceptable with a large directory.

Undefined Synchronization Behavior

After running `idsync linkusers`, you must resynchronize existing users (run `idsync resync`) or the resynchronization behavior will remain undefined.

User Resynchronization

Prior to starting synchronization for your network ensure that all existing users are synchronized between servers. User Resynchronization `idsync resync` enables an administrator to bulk resynchronize two directory sources. `Idsync resync` can create users and synchronize attributes, but it cannot synchronize passwords. (The one exception to password synchronization is invalidating the Sun ONE Directory Server password to force on-demand password synchronization in an Active Directory environment.)

- `idsync resync` can populate an empty Sun ONE Directory Server with existing Active Directory or NT domain users.
- After running `idsync linkusers` to link users in two existing directory sources, `idsync resync` can synchronize all user entry attribute values (other than passwords) in the two directories.

NOTE If there are existing Windows users, then the `idsync resync` command must be run before synchronization is started. If you do not want to synchronize existing users to Directory Server, then run it with the `-u` flag, which only updates the object cache and does not synchronize the Windows' entries to Directory Server. If you have existing Windows users, and you do not run `idsync resync`, then changes to these users may or may not be propagated, and depending on flow settings, they might even be automatically created in Directory Server. `idsync resync` must be run even if `idsync resync` was already run.

- If two directory sources become out of sync, then `idsync resync` can synchronize user entries.
- `Idsync resync` can "prime" the object cache database of the NT and Active Directory connectors. The object cache maintains a shadow copy of the Active Directory or NT SAM user entires.

Usage

The `idsync resync` command accepts the following arguments:

Table 6-3 `idsync resync` Usage

Option	Meaning
-h	Configuration directory hostname.
-p	Configuration directory port number.
-D	Directory Manager bind distinguished name.
-w	Directory Manager Bind password
-s	Suffix of the configuration directory.
-o	Controls the source of the `resync` operation. For example, if Windows is specified, then the attribute values in the Sun ONE Directory Server entries are set to the corresponding attribute values in the Windows directory source entry. The Default is Windows. If you run the `resync` command without the `-o` flag set, then the `resync` operation will update user entries in the Sun ONE Directory Server.
-q	Configuration password.

Option	Meaning
-l	Synchronized user list selection. Specifies individual synchronized user lists to synchronize. Multiple can be specified by repeating this option (e.g. `–l SUL1 –l SUL2`). If none are specified then all synchronized user lists are synchronized.
-c	Create Users. If a corresponding user is not found at the destination directory source, then the scripts automatically create the user entry. If this option is not supplied, then users are not created. This option should only be used with an empty directory or a directory with users that have already been linked using `idsync linkusers`.
-u	Update object cache. This option only updates the local cache of user entries for a Windows directory source. This prevents pre-existing Windows users from being created in the Sun ONE Directory Server. If this option is used, Windows user entries are not synchronized with Sun ONE Directory Server user entries. This option is only valid when the `resync` source is Windows.
-a	Specify an LDAP filter to control the synchronization of individuals or groups.
-i	Resets passwords of resynchronized Directory Server users (resynching to Sun only). After a Directory Server user’s password has been reset, the program will invoke on-demand password synchronization the next time that user logs on. This option has two possible values: • ALL_USERS: Resets the passwords of all resynchronized users. • NEW_USERS: Resets the passwords of users created during resynchronization.
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.

Option	Meaning
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: /var/Sun/MPS/slapd-<serverID>/secmod.db You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Example Usages

Some sample combination of command line options are shown. The standard `–h`, `-p`, `-D`, `-w`, `-s` options have been omitted for brevity.

1. Update the value of all Sun ONE Directory Server user attribute values with the values of the attributes in the Windows environment, but do not create users that cannot be found:

   ```
   idsync resync
   ```

2. The same as 1. except the passwords of all users are invalidated to force on-demand synchronization. This command is only valid in an AD only environment. In a mixed environment with both AD and NT domains, the AD SULs must be listed explicitly

   ```
   idsync resync -i ALL_USERS
   ```

3. The same as 1. except users that are not found are created in Sun ONE Directory Server and their passwords are invalidated to force on-demand synchronization. This command can be used to populate an empty Sun ONE Directory Server instance with existing Windows users.

   ```
   idsync resync -c -i NEW_USERS
   ```

4. Create all existing AD users in Sun ONE Directory Server for only the SUL_sales and SUL_finance SULs (but do not force on-demand synchronization)

   ```
   idsync resync -c SUL_sales SUL_finance
   ```

5. The same as 1. but only display and log what would occur (no users are modified or created):

   ```
   idsync resync -n
   ```

6. Synchronize all Sun ONE Directory Server users in Windows that have a last name (sn) of Smith:

   ```
   idsync resync -o Sun -a "(sn=Smith)"
   ```

7. Only update the object cache for Windows connectors to prevent existing user from being created in Directory Server. No users are actually synchronized:

   ```
   idsync resync -u
   ```

Logging

Events from a `idsync resync` operation are logged to the same logs as `linkusers` operations. They are logged to the `linking.log` file.

Starting and Stopping Synchronization

This section explains how to start and stop a Identity Synchronization for Windows instance. To start or stop synchronization:

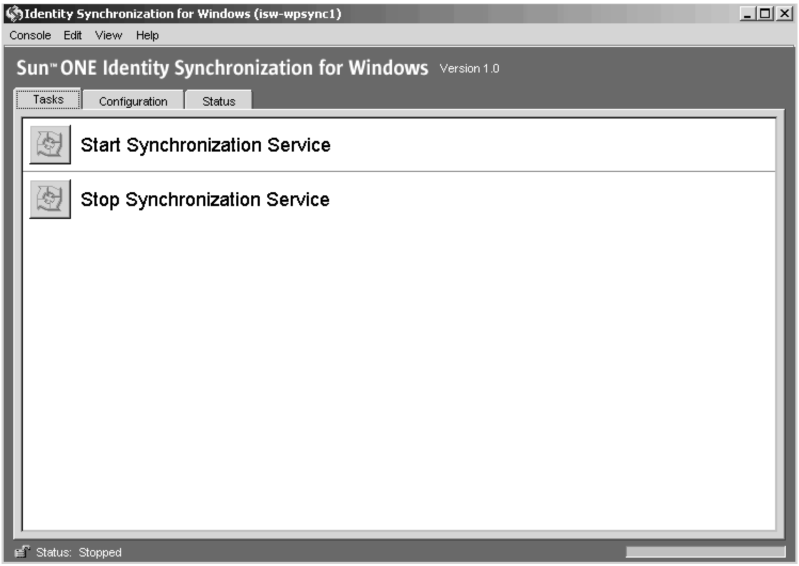
1. Access the console.

2. In the navigation tree, select the appropriate Identity Synchronization for Windows instance and press Open.

   Enter the configuration data password.

3. To stop synchronization service press the Tasks tab and then Stop.

4. To start synchronization service press the Tasks tab and then Start.


Identity Synchronization for Windows (isw-wpsync1)
Console Edit View Help
Sun™ ONE Identity Synchronization for Windows Version 1.0
Tasks
Configuration
Status
Start Synchronization Service
Stop Synchronization Service
Status: Stopped

Starting and Stopping Services

On Solaris, Identity Synchronization for Windows and the Sun ONE Message Queue are installed as daemons. They are started automatically when the system boots. The `/etc/init.d/isw` and `/etc/init.d/imq` scripts can be used to start and stop these daemons manually. For example,

- `/etc/init.d/isw start` starts all Identity Synchronization for Windows processes
- `/etc/init.d/isw stop stops` all Identity Synchronization for Windows processes

NOTE Pause 30 seconds after stopping the Identity Synchronization for Windows daemon/service before starting it again because it might take several seconds for a connector to cleanly shut itself down.

- `/etc/init.d/imq start` starts the Message Queue broker
- `/etc/init.d/imq stop` stops the Message Queue broker

On Windows, Identity Synchronization for Windows and the Sun ONE Message Queue are installed as Windows services. The “Sun ONE Identity Synchronization for Windows“and “iMQ. Broker” services are started up automatically when the system boots. To control them manually, use the Services control panel or the net command.

Chapter 7

Removing the Software

This section contains procedures for removing Identity Synchronization for Windows in the following sections:

- Planning for Uninstallation
- Uninstalling on Windows NT Platforms
- Uninstalling on Solaris and Windows 2000

Planning for Uninstallation

Before removing the software keep in mind the following points:

- Uninstall subcomponents before their associated connectors and all the connectors before core.
- Always uninstall the NT subcomponents before an NT connector.
- Always uninstall the Directory Server plugin subcomponent before the Directory Server connector.
- In replicated environments with replica's in addition to primary and secondary, the Directory Server Plugin subcomponents must also be uninstalled and those servers restarted.
- The Active Directory connector does not have any subcomponent to uninstall.
- The order in which you uninstall connectors does not matter.

- After uninstalling a Sun ONE Directory Server or Windows NT connector, additional steps must be followed if the connector is reinstalled on a different machine or using a different server port. In this case, all of its corresponding subcomponents must be uninstalled and reinstalled, and the Identity Synchronization for Windows daemon/service where core is installed must be restarted (see Starting and Stopping Services).
- You cannot uninstall core unless all the connectors and subcomponents on all systems have been uninstalled.
- On Windows 2000 and NT platforms, use runUninstaller.bat. This script is located in the isw-home directory. This batch file must be run as administrator.
- On Solaris environments, use runUninstaller.sh. This script is located in the isw-home directory. This script must be run as root user.

Uninstalling on Windows NT Platforms

Execute the following steps on each NT primary domain controller in your Identity Synchronization for Windows deployment.

Uninstalling NT Subcomponents

1. Use the runUninstaller.bat script on the desired NT Server. Find it in the isw-*hostname* directory.
2. At the Welcome screen press Next.
3. Select Windows NT Subcomponent to uninstall and press next.
4. Enter the Sun ONE Directory Server URL, administrator’s name, and password.
5. A summary window appears, listing all components to removed. Review the listing and press Back to make changes. Press Uninstall Now to remove the listed components.
6. Press Next to perform further uninstallation related tasks.
7. A summary window appears. Press Details to see the uninstallation log. Press Close to exit the program.
8. Reboot the NT system.

Uninstalling the NT Connector

9. Run the runUninstaller.bat script on the same NT Server. Find it in the isw-*hostname* directory.
10. At the Welcome screen press Next.
11. Select Core / Connectors and select the NT Connectors to uninstall.
12. Enter the Sun ONE Directory Server URL, administrator's name, and password.
13. A summary window appears, listing all components to removed. Review the listing and press Back to make changes. Press Uninstall Now to remove the listed components.
14. Press Next to perform further uninstallation related tasks.
15. A summary window appears. Press Details to see the uninstallation log. Press Close to exit the program.
16. When Prompted, Reboot the NT system.
17. Repeat all steps Uninstalling NT Subcomponents and Uninstalling the NT Connector on each NT server in your network.

NOTE If you are unable to run the connector uninstaller for a given connector for any reason (for example, if you lost the connector files during a hard drive failure), use the `resetconn` command.

This command resets the connector state in the Configuration Registry (CR) to *uninstalled* so that you can reinstall it elsewhere. `resetconn` is similar to other commands that access the CR, and it provides two options:

- **-e** <*name of directory source*>: Specifies the name of the directory source to be reset. (Connectors are identified in the installers by their directory source name.)
- **-n** (safe mode): Indicates whether the arguments specified for the command are correct without doing any work.

Example command:

```
idsync resetconn -h host -p 389 -D "cn=Directory Manager"
-w secret -s dc=central,dc=example,dc=com -q secret2 -e
dc=central,dc=example,dc=com
```

`resetconn` Output:

```
NOTICE: This program will reset the installation state to
UNINSTALLED for the Connector associated with the
specified DirectorySource ’dc=central,dc=example,dc=com’.

Changing the Connector to an UNINSTALLED state is a last
resort.  This is NOT meant to be used for uninstalling
connectors.  It is typically used if you lost a machine
with the connector on it and can not run the uninstaller.
Additionally, this program will rewrite the existing
configuration. This can be a lengthy process. Before
proceeding, you should stop the console, any running
installers, and all other system processes. You may want
to export the ou=Services tree in the Configuration
Registry to ldif as a backup.

Do you want to reset the installer settings for the
connector (y/n)?
```

Uninstalling on Solaris and Windows 2000

Your Solaris or Windows 2000 system may contain any or all of the following Identity Synchronization for Windows components:

- Active Directory connectors
- Directory Server connectors and plugins
- Core

Use `runUninstaller.sh` (Solaris) or `runUninstaller.bat` (Windows) to remove all connectors and subcomponents and then remove core (if installed). This section contains:

- Uninstalling Directory Subcomponent (plugin)
- Uninstalling Connectors
- Uninstalling Core

Uninstalling Directory Subcomponent (plugin)

1. Start the runUninstaller program. Find it in the isw-*hostname* directory.
2. At the Welcome screen press Next.
3. Select Sun ONE Directory Server Plugin and press next.
4. Enter the Sun ONE Directory Server URL, administrator's name, and password.
5. A summary window appears, listing all components to be removed. Review the listing and press Back to make changes. Press Uninstall Now to remove the listed components.
6. Press Next to perform further uninstallation related tasks.
7. When prompted restart the Directory Server where the plugin subcomponent was installed.
8. A summary window appears. Press Details to see the uninstallation log. Press Close to exit the program.
9. If the Directory Server subcomponent is the *only* Identity Synchronization for Windows component installed on the target host, then you can delete the isw-*hostname* folder.

10. Repeat Step 1 through Step 8 for each Directory Server Plugin subcomponent installed on a Windows 2000 server in your network.

Uninstalling Connectors

1. Start the runUninstaller program. Find it in the isw-*hostname* directory.
2. At the Welcome screen press Next.
3. Select Core and Connectors to uninstall and press next.
4. Enter the configuration directory URL, press Fetch Root Suffixes, and select the appropriate root suffix from the drop-down menu.
5. Enter the administrator's name and password for the configuration directory.
6. Select the connector(s) to be uninstalled.

NOTE The selected connectors *must* be present on the target host.

7. A summary window appears, listing all components to removed. Review the listing and press Back to make changes. Press Uninstall Now to remove the listed components.
8. Press Next to perform further uninstallation related tasks.
9. A summary window appears. Press Details to see the uninstallation log. Press Close to exit the program.
10. If there are no other installed on the target host, then you can safely remove the isw-*hostname* folder.
11. Repeat Step 1 through Step 10 for all hosts where connectors are installed.

Uninstalling Core

1. Start the runUninstaller program on the host where you have Identity Synchronization for Windows Core installed. Find it in the isw-*hostname* directory.
2. At the Welcome screen press Next.
3. Select Core and Connectors to uninstall and press next.

4. Enter the configuration directory URL, press Fetch Root Suffixes, and select the appropriate root suffix from the drop-down menu.

5. Enter the administrator's name and password for the configuration directory.

6. Select Core to be uninstalled.

7. A summary window appears, listing all components to removed. Review the listing and press Back to make changes. Press Uninstall Now to remove the listed components.

8. Press Next to perform further uninstallation related tasks.

9. A summary window appears. Press Details to see the uninstallation log. Press Close to exit the program.

10. Delete the isw-*hostname* folder.

Chapter 8

Troubleshooting

This chapter provides Identity Synchronization for Windows troubleshooting information.

It includes the following sections:

- Troubleshooting Checklist
- Troubleshooting Connectors
- Troubleshooting Components
- Troubleshooting Subcomponents
- Troubleshooting Sun ONE Message Queue
- Troubleshooting SSL Problems

Troubleshooting Checklist

NOTE Administrators: When you are debugging problems, adjust the logging level (as described in Logs and Status) to ensure the log reflects all events that may be causing problems.

Some events (such as the program failing to synchronize a user change because the user was not included in the SUL) are not included in a log file until you adjust the log level to FINE or higher. The log level should be left at INFO during all `idsync linkusers` and `idsync resync` operations.

1. Are there any problems reported in the central error.log?

`isw-`*`hostname`*`/logs/central/error.log`.

Almost all errors will be reported in this log file. More information on any error is usually available in the audit.log file. To ease correlation of related log entries, this file also includes all entries in the error log.

2. The Release Notes document many known issues. Is this problem explained there?
3. Was the installation performed on a clean machine? Problems might occur when this product is reinstalled if the uninstallation of the previous configuration was not complete. Please refer to Removing the Software for more instructions on how to clean up previous installations.
4. Was the core properly installed? If core installation completed successfully, then log files will exist in the `isw-`*`hostname`*`/logs/central/` directory.
5. Was the Directory Server running during resource configuration?
6. Is the core, including the Sun ONE Message Queue and the System Manager, currently running? On Windows, check for the appropriate service name. On Solaris, check for the appropriate daemon name. Use the `idsync printstat` command to verify that the Sun ONE Message Queue and System Manager are active.
7. Was a configuration saved successfully? If the `idsync printstat` command lists connectors, then a configuration was saved successfully.
8. Were all connectors installed? One connector must be installed for each directory source being synchronized.
9. Were all subcomponents installed? Sun ONE Directory Server and Windows NT connectors require subcomponents to be installed after the connector installation. The Sun ONE Directory Server plugin must be installed in each Sun ONE Directory Server replica.
10. Were post-installation procedures followed? The Sun ONE Directory Server must be restarted after the Directory Server plugin is installed. The Windows NT Primary Domain Controller must be rebooted after the Windows NT subcomponents are installed.
11. Was synchronization started either from the console or command line?
12. Are all connectors currently running?
13. Verify that all connectors are in the SYNCING state using the console or `idsync printstat`.
14. Are the directory sources being synchronized currently running?

15. Verify using the console that modifications and/or creates are synchronized in the expected direction(s).

16. If synchronizing users that existed in only one directory source, were these users created in the other directory source using the `idsync resync` command

NOTE	You must run `idsync resync` whenever there are existing users (even after running `idsync linkusers`). If you do not resynchronize existing users, resynchronization behavior remains undefined.

17. If synchronizing users that existed in both directory sources, were these users linked using the `idsync linkusers` command?

18. If user creates fail from Active Directory or Windows NT to the Sun ONE Directory Server, verify that all mandatory attributes in the Sun ONE Directory Server objectclass are specified as creation attributes and values for the corresponding attributes are present in the original user entry.

19. If synchronizing creates from Directory Server to Windows NT and the user creation succeeded, but the account is unusable, verify that the user name does not violate Windows NT requirements.

    For example, if you specify a name that exceeds the maximum allowable length for Windows NT, the user will be created on NT but will remain unusable and uneditable until you rename the user (User -> Rename).

20. For the Windows NT SAM Change Detector subcomponent to be effective, you must turn on the NT audit log. Under Start > Programs > Administrative Tools > User Manager, select Policies > Audit Policies.
    Select Audit These Events and then both the Success and Failure boxes for User and Group Management.

    Under Event Log Settings in the Event Viewer>Event Log Wrapping, select Overwrite Events as Needed.

21. Are the users that fail to synchronize within a Synchronization User List? I.e. do they match the base DN and filter of a Synchronization User List? In deployments that include Active Directory, on-demand password synchronization fails silently if the Sun ONE Directory Server entry is not in any Synchronization User List. This most often occurs because the filter on the Synchronization User List is incorrect.

22. Were the synchronization settings changed? If the synchronization settings changed from only synchronizing users from Active Directory to the Sun ONE Directory Server to synchronizing users from the Sun ONE Directory Server to Active Directory, then the Active Directory SSL CA certificate must be added to the connector's certificate database. The `idsync certinfo` command reports what SSL certificates must been installed based on the current SSL settings.

23. Are all host names properly specified and resolvable in DNS? The Active Directory domain controller should be DNS-resolvable from the machine where the Active Directory connector is running and the machine where the Sun ONE Directory Server plugin is running.

24. Does the IP address of the Active Directory domain controller resolve to the same name that the connector uses to connect to it?

25. Does the source connector detect the change to the user? Use the central audit.log to determine if the connector for the directory source where the user was added or modified detects the modification.

26. Does the destination connector process this modification?

27. Are multiple Synchronization User Lists configured? If so, are these in conflict? More specific Synchronization User Lists should be ordered before less specific ones using the console.

28. If flow is set to bidirectional or from Sun to Windows and there are Active Directory data sources in your deployment, are the connectors configured to use SSL communication?

29. If memory problems are suspected on Solaris environments check the processes. To view which components are running as different processes, enter

    ```
    /usr/ucb/ps -gauxwww | grep com.sun.directory.wps
    ```

    The output gives the full details including the ID of connectors, system manager and central logger. This can be useful to see if any of the processes are consuming excessive memory.

30. If you are creating or editing the Sun ONE Directory source, and the Directory Server does not display in the Choose a known server drop-down list, check that the Directory Server is running. The Directory Server must be running to appear in the drop down list of available hosts.

    If the server in question is down temporarily, type the host and port into the Specify a server by providing a hostname and port field.

31. Do you receive the following error while running uninstaller program?

```
./runInstaller.sh
```

```
IOException while making /tmp/SolarisNativeToolkit_5.5.1_1
executable:java.io.IOException: Not enough space
```

```
java.io.IOException: Not enough space
```

Increase the size of the swap file mounted at /tmp.

Troubleshooting Connectors

How to determine the ID of a connector managing a directory source

Using the central logs

Determine the connector IDs of the directory sources being synchronized by looking in the central audit log. At startup, the central logger logs the IDs of each connector and the directory source that it manages. Look for the last instance of the startup banner for the most recent information. For example, in the following log message there are two connectors: CNN101 is a Sun Directory connector that manages dc=airius,dc=com, and CNN100 is an Active Directory connector that manages the airius.com domain.

```
[2003/03/19 00:00:00.722 -0600] INFO    16     "System Component
Information:    SysMgr_100 is the system manager (CORE);  console is
the Product Console User Interface;  CNN101 is the connector that
manages [dc=airius,dc=com (ldap://host1.airius.com:389)];  CNN100 is
the connector that manages [airius.com
(ldaps://host2.airius.com:636)];"
```

Using idsync printstat

The connector IDs and status are also available from the idsync printstat command. A sample output of this command is shown below.

```
Connector ID: CNN100
   Type:     Active Directory
   Manages:  airius.com (ldaps://host2.airius.com:636)
   State:    READY

Connector ID: CNN101
```

```
   Type:     Sun ONE Directory
   Manages:  dc=airius,dc=com (ldap://host1.airius.com:389)
   State:    READY

Sun ONE Message Queue Status:  Started

Checking the System Manager status over the Sun ONE Message Queue.

System Manager Status:  Started

SUCCESS
```

How to determine a connector's current state.

Determine the current state of the connectors involved in the synchronization. This can be done using the status pane in the console, the `idsync printstat` command as shown above, or by looking in the central `audit.log`. Search for the last message in the `audit.log` that reports the state of the connector. For example, in this log message we see that connector CNN101 is in the READY state.

```
[2003/03/19 10:20:16.889 -0600] INFO    13  SysMgr_100 host1
"Connector [CNN101] is now in state "READY"."
```

Table 8-1 Connector State Meanings

State	Meaning
UNINSTALLED	The connector has not be installed.
INSTALLED	The connector has been installed, but it has not received its configuration.
READY	The connector has been installed and has received its configuration, but it has not started to synchronize.
SYNCING	The connector has been installed, has received its configuration, and has attempted to start synchronizing.

What to do if the connector is in the UNINSTALLED state.

Install the connector.

What to do if the connector is in the INSTALLED state.

If a connector remains in the installed state for a long period of time, then most likely it is not running, or it is unable to communicate with the Sun ONE Message Queue.

At the machine where the connector was installed, look in the connector's logs (`audit.log` and `error.log`) for potential errors. If the connector cannot connect to the Sun ONE Message Queue, then that error will be reported here. If this is the case, see "Troubleshooting Sun ONE Message Queue" on page 138 for possible causes.

If the most recent messages in the audit log are old, then perhaps the connector is not running. See "Troubleshooting Components" on page 134.

What to do if the connector is in the READY state.

A connector remains in the READY state until synchronization has been started and all of its subcomponents have been installed and have connected to the connector. If synchronization has not been started, then start it using the console or command line utility.

If synchronization has been started, but a connector does not enter the SYNCING state, then there is likely a problem with subcomponent. See "Troubleshooting Subcomponents" on page 136.

What to do if the connector is in the SYNCING state.

If all connectors are in the SYNCING state, but modifications are not being synchronized, then verify that the synchronization settings are correct:

- Using the console, verify that modifications and/or creates are synchronized in the expected direction (e.g. from Windows to the Sun ONE Directory Server).
- Using the console, verify that the attribute being modified is a synchronized attribute (note: passwords are always synchronized). If created user entries are not being synchronized, then verify that user creation flow is enabled in the console.
- Does the source connector detect the change to the user? Use the central `audit.log` to determine if the connector for the directory source where the user was added or modified detects the modification. Does the destination connector process this modification?

Troubleshooting Components

On Windows:

Using the Service control panel, check that the "Sun ONE Identity Synchronization for Windows" service is started. If it is not started, then Identity Synchronization for Windows is not running on that machine, and should be started. If the service is started, then verify using the Task Manager that `pswwatchdog.exe` is running and that the expected number of `java.exe` processes are running:

- One for the Sun ONE Message Queue broker only if the core is installed
- One for the System Manager only if the core is installed
- One for the Central Logger only if the core is installed
- One for each Connector installed on that machine

NOTE There might be other active java processes, such as the Sun ONE Directory Server console. If `pswwatchdog.exe` is not running, then restart the "Sun ONE Identity Synchronization for Windows" service. If it is running but the expected number of `java.exe` processes are not running, then see "Examining WatchList.properties" on page 135 to verify that all components were installed properly.

On Solaris:

The command **`/usr/ucb/ps -auxww | grep com.sun.directory.wps`** will list all of the Identity Synchronization for Windows processes running. This table shows which processes should be running.

Table 8-2 Identity Synchronization for Windows Processes

Java Process Class Name	Component	When Present
com.sun.directory.wps.watchdog.server.WatchDog	System Watchdog	always
com.sun.directory.wps.centrallogger.CentralLoggerManager	Central Logger	only where core is installed

com.sun.directory.wps.manager.SystemManager	System Manager	only where core is installed
com.sun.directory.wps.controller.AgentHarness	Connector	one for each connector installed

If the expected number of processes are not running, then issue the following commands to restart all Identity Synchronization for Windows processes.

```
# /etc/init.d/isw stop
```

```
# /etc/init.d/isw start
```

If the WatchDog process is running, but the expected number of `java.exe` processes are not running, then see the "Examining WatchList.properties" section below to verify that all components were installed properly.

Like other system components, the Sun ONE Directory Server plugin sends log records over the bus that are managed by the central logger for end-user viewing. However, the plugin also logs some messages that may not show up over the bus (for instance when the subcomponent cannot contact the connector). In this case the log messages only show up in the plugin's log directory on the file system, which should look something like `<server root>/isw-<host>/logs/SUBC<id>`.

Since the plugin runs in process with the directory server, there could potentially be a problem for the plugin's ability to write into its log directory. This happens if the directory server runs as a different user than the owner of the log directory. In this case, it may be necessary to give the plugin permission explicitly by changing the directories permission or owner using native operating system tools.

Examining WatchList.properties

On each machine where a Identity Synchronization for Windows component is installed, the `isw-<`*`machine-name`*`>/resources/WatchList.properties` file enumerates the components that should run on that machine. The `process.name[n]` properties name the components that should be running.

On machines where core is installed, `WatchList.properties` will include entries for the Central Logger and System Manager:

```
process.name[1]=Central Logger
…
process.name[2]=System Manager
…
```

On machines where connectors are installed, `WatchList.properties` will include a separate entry for each connector. The `process.name` property is the connector ID:

```
process.name[3]=CNN100
```

…

```
process.name[4]=CNN101
```

…

If there is a mismatch between the entries in `WatchList.properties` and the actively running processes, then restart the Identity Synchronization for Windows daemon or service.

If there are fewer than expected entries in `WatchList.properties` (e.g. only one connector entry even though two were installed), then examine the installation logs for possible installation failures. On Solaris, these logs are in `/var/sadm/install/logs/` and on Windows, they are in the `%TEMP%` directory.

Troubleshooting Subcomponents

1. Have all subcomponents been installed?

   Subcomponent installation must be done after the connector is installed:

- For Active Directory connectors, no subcomponents are installed.
- For Sun ONE Directory connectors, the plugin must be installed at the Sun ONE Directory Server being synchronized.
- For Windows NT connectors, the Windows change detector and password filter plugins must be installed on the primary domain controller for each Windows NT domain being synchronized. These two subcomponents are installed together after the Windows NT connector has been installed.

.

NOTE For the Windows NT SAM Change Detector subcomponent to be effective, you must turn on the NT audit log. Under Start > Programs > Administrative Tools > User Manager, select Policies > Audit Policies.
Select Audit These Events and then both the Success and Failure boxes for User and Group Management.

Under Event Log Settings in the Event Viewer>Event Log Wrapping, select Overwrite Events as Needed.

2. Have the subcomponent post installation steps been followed?

After the Directory Server plugin has been installed at the Sun ONE Directory Server, the server must be restarted. After the NT change detector and password filter have been installed on the primary domain controller, the server must be rebooted.

3. Are the subcomponents running?

Is the Sun ONE Directory Server where the plugin was installed running? Is the Primary Domain Controller where the Change Detector and Password Filter were installed running?

4. Have the subcomponents established a network connection to the connector?

On the machine where the connector is running, verify that the connector is listening for the subcomponent's connection by running `netstat –n –a`. The following examples show the results of this command for three different scenarios. (The connector was configured to listen on port 9999.)

a. The connector is listening for incoming connections, and the subcomponent has successfully connected:

```
> netstat –n –a | grep 9999
      *.9999               *.*    0   0 65536    0 LISTEN
12.13.1.2.44397 12.13.1.2.9999  73620 0 73620    0 ESTABLISHED
12.13.1.2.9999  12.13.1.2.44397 73620 0 73620    0 ESTABLISHED
```

This is the expected result.

b. The connector is listening for incoming connections, but the subcomponent has not connected:

```
# netstat –n –a | grep 9999
      *.9999               *.*    0   0 65536    0 LISTEN
```

After verifying that the subcomponent is running, examine the subcomponent's local logs for potential problems.

c. The connector is not listening for incoming connections:

```
# netstat –n –a | grep 9999
<no output>
```

Verify that the correct port number was specified. Verify that the connector is running and is in the READY state. Examine the connector's local logs for potential problems.

Troubleshooting Sun ONE Message Queue

Verify that the Sun ONE Message Queue broker is running. Issuing a `telnet` command to the machine and port where the Sun ONE Message Queue broker is running will return a list of the active Message Queue services:

```
# telnet localhost 7676
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
101 psw-broker 3.0.1
cluster tcp CLUSTER 32914
admin tcp ADMIN 32912
portmapper tcp PORTMAPPER 7676
ssljms tls NORMAL 32913
jms tcp NORMAL 32911
.
```

```
Connection closed by foreign host.
```

If the “ssljms tcp NORMAL” service is not listed in the output, then examine the Sun ONE Message Queue logs for potential problems. If the core was installed on Solaris, then the Sun ONE Message Queue broker’s log is `/var/imq/instances/psw-broker/log/log.txt.` Otherwise, if the core was installed on Windows, then the broker’s log is:

<*installation-root*>\isw-machine-name\imq\var\instances\isw-broker\log\log.txt.

If `telnet` command fails, then either the broker is not running or the wrong port was specified. Verify the port number by checking the port number in the broker’s log. The broker’s port is specified in the following line

```
[13/Mar/2003:18:17:09 CST] [B1004]: “Starting the portmapper service
using tcp [ 7676, 50 ] with min threads 1 and max threads of 1”
```

If the broker is not running, then it can be started on Solaris by running `/etc/init.d/imq start` and on Windows by starting the `iMQ Broker` Windows service.

Troubleshooting Broker Configuration Directory Communication

The Sun ONE Message Queue broker authenticates clients against the Sun ONE Directory Server that stores the Identity Synchronization configuration. If the broker is unable to connect to this directory server, no clients will be able to connect to the Sun ONE Message Queue, and the broker log will mention some javax.naming exception, such as “javax.naming.CommunicationException” or “javax.naming.NameNotFoundException”. If this occurs, do the following

- Verify that all `imq.user_repository.ldap properties` in `/var/imq/instances/isw-broker/props/config.properties` have the correct values. If any of these is incorrect, stop the Sun ONE Message Queue broker, correct and save the file, and restart the broker. The directory server host name must be resolvable from the broker’s machine.
- Verify that the `imq.user_repository.ldap.password` property in `/etc/imq/passfile` is correct.
- In some cases, the broker cannot search for entries if the root suffix contains spaces.

Troubleshooting Broker Memory Settings

During normal operation, the Sun ONE Message Queue broker consumes a modest amount of memory. However during `idsync linkusers` and `idsync resync` operations, the broker’s memory requirements increase. If the broker reaches its memory limit, undelivered messages will accumulate, the `idsync linkusers` or `idsync resync` operation will slow down dramatically or stop completely, and the Identity Synchronization system might be unresponsive after this. When the broker enters a low memory state, the following messages will appear in its log

```
[03/Nov/2003:14:07:51 CST] [B1089]: In low memory condition, Broker
is attempting to free up resources
```

```
[03/Nov/2003:14:07:51 CST] [B1088]: Entering Memory State [B0024]:
RED  from previous state [B0023]: ORANGE - current memory is
1829876K, 90% of total memory
```

To avoid this situation,

- Increase the broker’s memory limit to 1 or 2 GB. This process is explained in the Release Notes.

- During an `idsync linkuser` and `idsync resync` operation, keep the log level at INFO. Changing the log level to FINE or higher increases the load at the broker as more log messages are sent to the central logger.

- Run the `idsync linkuser` or `idsync resync` operation for a single Synchronization User List at a time.

If the broker does run out of memory, follow these steps to recover:

1. Verify that the broker has a backlog of undelivered messages by examining its persistent message store. On Solaris, the broker's persistent message store is in the `/var/imq/instances/psw-broker/filestore/message/` directory, and on Windows it is in the`<`*`installation-root`*`>\isw-machine-name\imq\var\instances\isw-broker\filestore\message\` directory. Each file in this directory contains a single undelivered message. If there are more than 10000 files in this directory, then the broker has a backlog of messages.[1] Otherwise, there is another problem with the broker.

2. The backlog of messages are most likely only log files related to an `idsync linkuser` and `idsync resync` operation and can safely be removed.

3. Stop the Sun ONE Message Queue broker as described in Starting and Stopping Services.

4. Remove all files in the persistent message store. This can most easily be done by recursively removing the `message/` directory and then recreating it.

5. Restart the Sun ONE Message Queue broker.

6. Follow the steps above to make sure the broker does not run out of memory again.

Troubleshooting SSL Problems

When diagnosing problems with SSL, also see the Configuring Security, which describes how to setup SSL between components in Sun ONE Identity Synchronization. This section contains:

- SSL Between Core Components

- SSL between Connectors and the Sun ONE Directory Server or Active Directory

- SSL between the Sun ONE Directory Server Plugin and Active Directory

[1] Even if all messages have been delivered, the broker might maintain up to 10000 message files to avoid the performance penalty of creating and deleting files.

SSL Between Core Components

The Identity Synchronization for Windows installer cannot verify that the SSL port provided during core installation is correct. If you incorrectly type the SSL port during core installation, then the core components will not be able to communicate properly. You may not notice any problem till you try to save the configuration for the first time. The console will alert you with the following warning: “The configuration was successfully saved, however, the System Manager could not be notified of the new configuration.”

The system manager logs will have the following entry:

```
[10/Nov/2003:10:24:35.137 -0600] WARNING 14   example  "Failed to
connect to the configuration registry because "Unable to connect:
(-5981) Connection refused by peer.".  Will retry shortly."
```

In this situation, uninstall the core and install it again with the correct SSL port number.

SSL between Connectors and the Sun ONE Directory Server or Active Directory

If a connector is unable to connect over SSL to the Sun ONE Directory Server or Active Directory, then this message will appear in the central error log:

```
[06/Oct/2003:14:02:48.911 -0600] WARNING 14  CNN100 host1  "failed
to open connection to ldaps://host2.airius.com:636."
```

Untrusted Certificates

More information will be available in the central audit log. For example, if the LDAP server’s SSL certificate is not trusted this message will be logged

```
[06/Oct/2003:14:02:48.951 -0600] INFO    14  CNN100 host1  "failed
to open connection to ldaps://host2.airius.com:636, error(91):
Cannot connect to the LDAP server, reason: SSL_ForceHandshake
failed: (-8179) Peer’s Certificate issuer is not recognized."
```

In most situations, the CA certificate has not been added to the connector’s certificate database. This can be confirmed by running the `certutil` program that ships with the Sun ONE Directory Server.[1] In this example, the certificate database contains no certificates:[2]

[1] Before running this command on Solaris, the <installation-root>/lib directory must be added to the LD_LIBRARY_PATH environment variable.

```
# /usr/sunone/servers/shared/bin/certutil -L -d /usr/sunone/servers/
isw-host1/etc/CNN100

Certificate Name                                  Trust Attributes

p    Valid peer
P    Trusted peer (implies p)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

In the following example, the certificate database contains only the Active Directory CA certificate:

```
# /usr/sunone/servers/shared/bin/certutil -L -d /usr/sunone/servers/
isw-host1/etc/CNN100

Certificate Name                                 Trust Attributes

airius.com CA                                    C,c,

p    Valid peer
P    Trusted peer (implies p)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

As shown here, the trust flags of the CA certificate must be “C,,”. If the certificate exists and the trust flags are set properly, but the connector still cannot connect, then first verify that the connector was restarted after adding the certificate, and then use the `ldapsearch` command that ships with the Sun ONE Directory to help diagnose the problem. If `ldapsearch` does not accept the certificate, then neither will the connector. For example, `ldapsearch` can reject certificates if they are not trusted

```
# /usr/sunone/servers/shared/bin/ldapsearch -Z -P /usr/sunone/
servers/isw-host1/etc/CNN100 -h host2 -b "" -s base
"(objectclass=*)"
ldap_search: Can’t contact LDAP server
    SSL error -8179 (Peer’s Certificate issuer is not recognized.)
```

[2] The default certificate databases for the Sun ONE Directory Server and Windows NT connectors include two certificates, saint-cert100 and saintRootCA. These certificates are not used in this release.

The `-P` option directs `ldapsearch` to use connector CNN100's certificate database for SSL certificate validation. After the correct certificate is added to the connector's certificate database, verify that `ldapsearch` accepts the certificate, and then restart the connector.

Expired Certificates

If the server's certificate has expired, this message will be logged

```
[06/Oct/2003:14:06:47.130 -0600] INFO    20  CNN100 host1  "failed
to open connection to ldaps://host2.airius.com:636, error(91):
Cannot connect to the LDAP server, reason: SSL_ForceHandshake
failed: (-8181) Peer's Certificate has expired."
```

In this case, the server must be issued a new certificate.

SSL between the Sun ONE Directory Server Plugin and Active Directory

By default the Sun ONE Directory Server does not communicate with Active Directory over SSL when performing on-demand password synchronization. If the default is overridden to protect this communication with SSL, then the Active Directory CA certificate must be added to the Sun ONE Directory Server certificate database of each master replica as described in Configuring Security. If this certificate is not added, users will fail to bind to the Sun ONE Directory Server with the error "DSA is unwilling to perform.", and the plugin's log (e.g. isw-hostname/logs/SUBC100/pluginwps_log_0.txt) will report

```
[06/Nov/2003:15:56:16.310 -0600] INFO  td=0x0376DD74 logCode=81
ADRepository.cpp:310    "unable to open connection to Active
Directory server at ldaps://host2.airius.com:636, reason: "
```

In this situation, the Active Directory CA certificate must be added to the directory server's certificate database and the directory server restarted.

Chapter 9

Logs and Status

Identity Synchronization for Windows logs information into an Audit Log and Error Log. The Audit Log provides information for day-to-day operations which include error conditions that are contained in the Error Log. The Error Log essentially acts as a filter such that only error entries are displayed.

This chapter includes the following sections:

- Setting Log Levels
- Viewing the Audit or Error File
- Understanding Logs
- Understanding Logs

Setting Log Levels

1. Access the Console.
2. In the navigation tree, expand the relevant domain, host, and server group tree nodes where Identity Synchronization for Windows has been installed.
3. Select the appropriate Identity Synchronization for Windows instance and press Open.
4. Enter the configuration password.
5. At the console press the Configuration tab.

6. In the navigation tree, select Log. The Log Files configuration window (Figure 9-1) contains the following:

   - **Write logs to file.** Select this option to write logs to a file on the core host. After selecting this you may:

   - Select the Default log directory and file.

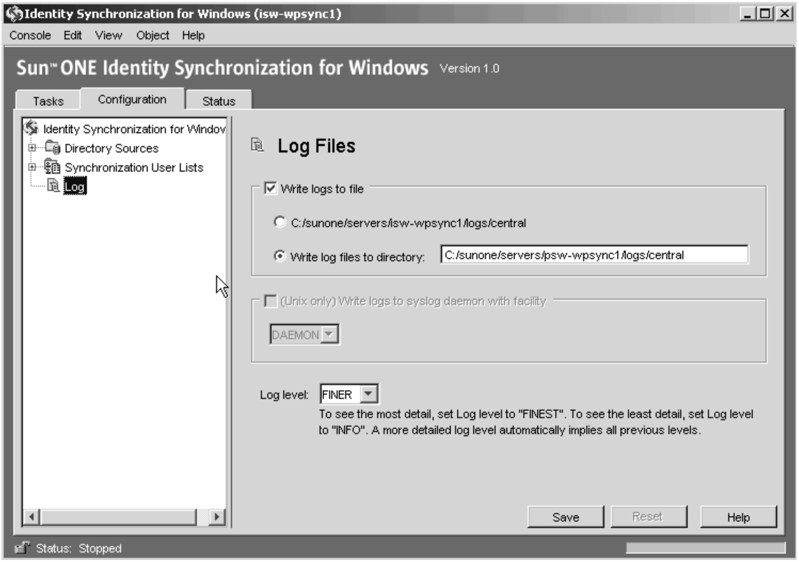
   - Specify a path and filename for the log file.

NOTE The Console does not verify whether a specified log file location actually exists. Consequently, there is no indication that you specified and saved a nonexistent log location until you try to view the logs. After several attempts to view the logs, a message displays to report the Console's inability to find logs at the specified location.

   - (Solaris Only) **Write logs to syslog daemon with facility**. Select this option if Identity Synchronization for Windows resides on a Solaris platform. Choose from the drop down menu the method to write the log. Daemon is the default.

   - **Log Level**. Choose from the drop down menu the level of logging for the system. The choices run from INFO to FINEST. See "Log Format" on page 151 for more information.

Figure 9-1 Log Files Configuration



Identity Synchronization for Windows (isw-wpsync1)
Console
Edit
View
Object
Help
Sun™ ONE Identity Synchronization for Windows Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
Synchronization User Lists
Log
Save
Reset
Help
Status: Stopped

Log Files

☑ Write logs to file

○ C:/sunone/servers/isw-wpsync1/logs/central

● Write log files to directory: C:/sunone/servers/psw-wpsync1/logs/central

□ (Unix only) Write logs to syslog daemon with facility

DAEMON

Log level: FINER

To see the most detail, set Log level to "FINEST". To see the least detail, set Log level to "INFO". A more detailed log level automatically implies all previous levels.

7. If desired, select Write logs to file and then select either the default log file or specify a path and file.
8. If logging on a Solaris system, check the box next to (Solaris Only) Write logs to syslog daemon with facility. Then select the facility from the drop down menu. Daemon is the default.
9. Select from the Log Level drop down menu the appropriate level.

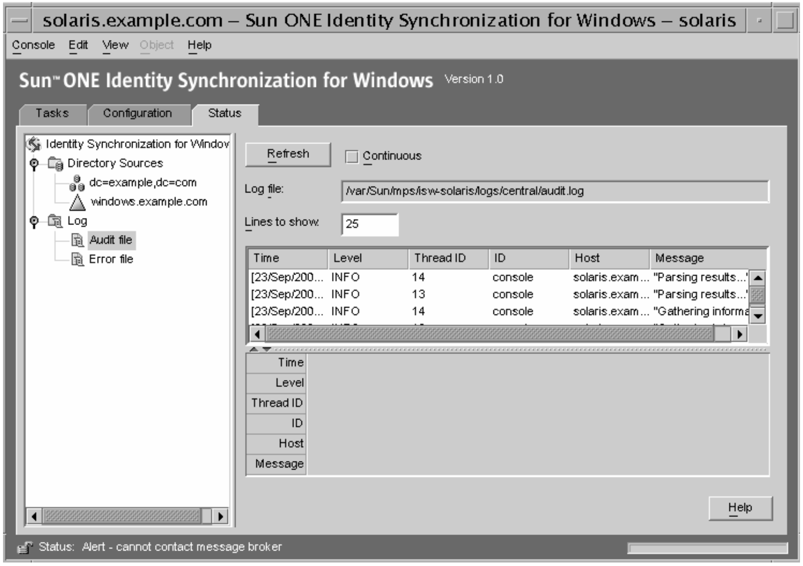
   Refer to "Log Levels" on page 152 for details on log level values.
10. Press Save to create the log file with the selected options.

Viewing the Audit or Error File

1. Access the Console.
2. In the navigation tree, expand the relevant domain, host, and server group tree nodes where Identity Synchronization for Windows has been installed.
3. Select the appropriate Identity Synchronization for Windows instance and press Open.
4. Enter the configuration password.
5. At the Identity Synchronization for Windows console press the Status tab.
6. In the navigation tree, expand Log and select Audit or Error file.

   The Audit file contents are displayed as shown in Figure 9-2.

Figure 9-2 The Audit Log


solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris
Console
Edit
View
Object
Help
Sun™ ONE Identity Synchronization for Windows
Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
dc=example,dc=com
windows.example.com
Log
Audit file
Error file
Refresh
Help

□ Continuous

Log file: /var/Sun/mps/isw-solaris/logs/central/audit.log

Lines to show: 25

Time	Level	Thread ID	ID	Host	Message
[23/Sep/200...	INFO	14	console	solaris.exam...	"Parsing results..."
[23/Sep/200...	INFO	13	console	solaris.exam...	"Parsing results..."
[23/Sep/200...	INFO	14	console	solaris.exam...	"Gathering informa

Time	
Level	
Thread ID	
ID	
Host	
Message	

Status: Alert - cannot contact message broker

The Audit or Error Status tab includes the following:

- **Refresh.** Press refresh to load the latest audit or error information.
- **Continuous.** Check Continuous to constantly load the latest audit or error information.
- **Log File:** Log File is the full path name of the audit or error log being read; for example:

```
C:\Program Files\Sun\MPS\psw-hostname\logs\central\audit.log
```

- **Lines to show:** Enter the number of audit or error entries to display. The default is 25.
- **Time/Date.** This is the time and date of the log entry.
- **Level.** This is the priority of this entry.
- **Thread ID.** This is the Java thread ID of the function generating the event.
- **Connector ID.** This is the connector issuing the event.
- **Host.** This is the fully qualified domain name of the host generating the event.
- **Message.** Information associated with the event.

Understanding Logs

Identity Synchronization for Windows has several log files. The central logs are the primary logs to monitor, but each component also has local logs, which can be used to diagnose problems with the connector if it cannot log to the central logger.

NOTE The number of each type of log file grows one per day indefinitely. Save or delete old logs to prevent running out of disk space.

Central Logs

Logs from all Identity Synchronization for Windows components are aggregated by the central logger. The central logs are the primary logs to monitor. As long as components can access the Sun ONE Message Queue, all error and audit messages will be logged here. These centralized logs, which include messages from all components, are located in the following directory on the machine where core is installed:

`<`*`installation-root`*`>/isw-machine-name/logs/central/`

The specific logs are

Table 9-1 Identity Synchronization for Window Log Types

Log Name	Description
`error.log`	Warning and Severe messages are reported here.
`audit.log`	A superset of `error.log` that includes messages about each synchronization event.
`linking.log`	Messages generated from `linkusers` and `resync` commands are reported here.

Each central log also includes information on each component ID. For example,

```
[2003/03/14 14:48:23.296 -0600] INFO    13     "System Component
Information:    SysMgr_100 is the system manager (CORE);  console is
the Product Console User Interface;  CNN100 is the connector that
manages [airius.com (ldaps:// server1.airius.com:636)];  CNN101 is
the connector that manages [dc=airius,dc=com (ldap://
server2.airius.com:389)];"
```

Local Component Logs

Each connector, the System Manager, and the Central Logger have the following local logs:

Table 9-2 Local Logs

Log Name	Description
`error.log`	Warning and Severe messages are reported here. These messages are also written to the central `error.log`.
`audit.log`	A superset of `error.log` that includes messages about each synchronization event. These messages are also written to the central audit.log.

These are located in the following subdirectories:

`<`*`installation-root`*`>/isw-machine-name/logs/`

The `sysmgr` and `clogger100` (central logger) directories are on the machine where core is installed.

These logs are rotated daily by moving the current log to a log file that includes the date as follows:

```
audit_2003_03_24.log
```

Local Subcomponent Logs

The following subcomponents also have local logs:

- Directory Server plugin
- NT Change Detector DLL
- Password Filter

These are located in the `SUBC1XX` (e.g. `SUBC100`) subdirectories of:

`<`*`installation-root`*`>/psw-machine-name/logs/ directory.`

These are limited to 1 MB in size, and the last 10 logs are kept.

Action ID

An action encompasses a single synchronization event. Log messages about the same action can be identified using the action's unique ID in the log.

Log messages for these actions also include a sequence number, which is increased for each log message. These sequence numbers can be used to order log messages which might arrive out of order at the central logger. In the following example CNN101-F3EE4A69F5-64 is the action ID and 6 is the sequence number of this log message:

```
Action ID=CNN101-F3EE4A69F5-64, SN=6
```

Log Format

Each log message includes the following information:

```
Time, Log Level, Thread ID, Component ID, Machine name, Log Message,
[Action ID]
```

For example,

```
[02/Oct/2003:15:30:55.609 -0600] INFO   27  CNN101 server1 "Action
processed successfully." (Action ID=CNN101-F3EE4A69F5-64, SN=6)
```

Log Levels

Log levels, which are included in each log message, are used to indicate the severity and verbosity of a log message. The log levels used in Identity Synchronization for Windows are:

Table 9-3 Log Levels

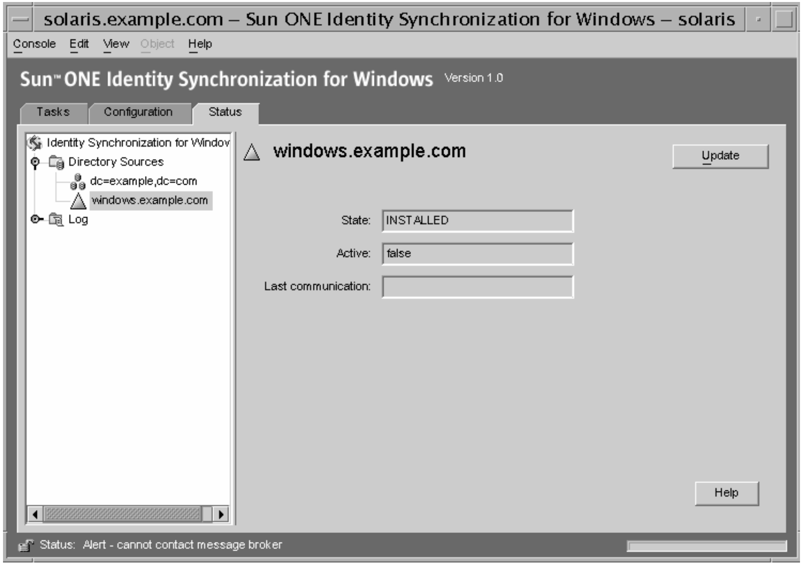
Log Level	**Description**
INFO	These messages allow the administrator to observe that the system is running correctly. They will include a minimum amount of information about each action, such as when it is detected and when the synchronization occurs. These messages are always logged to the audit log.
FINE	These messages contain more information about an action as it travels through the system.
FINER	These messages contain even more information about an action as it travels through the system. Turning the logging level to FINER for all components may impact performance.
FINEST	These messages contain the most information about an action as it travels through the system. Turning the logging level to FINEST for all components may significantly impact performance.

Viewing Directory Source Status

1. Access the Console.
2. In the navigation tree, expand the relevant domain, host, and server group tree nodes where Identity Synchronization for Windows has been installed.
3. Select the appropriate Identity Synchronization for Windows instance and press Open.
4. Enter the configuration password.
5. At the Identity Synchronization for Windows console press the Status tab.
6. In the navigation tree, expand Directory Source and select the appropriate source.

   The directory source contents are displayed as shown in Figure 9-3.

Figure 9-3 Directory Source Status


solaris.example.com – Sun ONE Identity Synchronization for Windows – solaris
Console
Edit
View
Object
Help
Sun™ ONE Identity Synchronization for Windows
Version 1.0
Tasks
Configuration
Status
Identity Synchronization for Windov
Directory Sources
dc=example,dc=com
windows.example.com
Log
Update
Help

windows.example.com

State: INSTALLED

Active: false

Last communication:

Status: Alert - cannot contact message broker

NOTE When viewing the Directory Source status you are essentially viewing the status of the connector associated with that Directory Source.

The Directory Source Status tab includes the following:

- **Update.** Press Update to refresh the information in this window.
- **State.** State reflects the current state of the directory source. Valid states include:
  - **Uninstalled.** The connector has not been installed.
  - **Installed.** The connector has been installed, but is not ready for synchronization yet. It has not received its runtime configuration yet.
  - **Ready.** The connector is ready for synchronization, but it is currently not synchronizing any objects yet.
  - **Syncing.** The connector is synchronizing objects.
- **Active**. Active notifies whether the directory source is active or down.

- **Last Communication.** This is the time of the last response from this directory source’s connector.

Chapter 10

Configuring Security

This chapter contains the following items:

- Security Overview
- Hardening your Security
- Securing Replicated Configurations
- Using idsync certinfo
- Enabling SSL in Directory Server
- Enabling SSL in the Active Directory Connector
- Adding the Active Directory Certificate to the Directory Server
- Adding the Directory Server Certificate to the Directory Server Connector

Some of the information in this section is written with an assumption that you are familiar with the basic concepts of public-key cryptography and Secure Sockets Layer (SSL) protocol, and understand the concepts of intranet, extranet, and the Internet security and the role of digital certificates in an enterprise. If you are new to these concepts, please refer to the security-related appendixes of the manual, *Managing Servers with iPlanet Console 5.0.*

Security Overview

Passwords are sensitive information; therefore, Identity Synchronization for Windows takes security precautions to ensure that user and administrative password credentials used to access the directories being synchronized are not compromised. This section covers the following security methodologies:

- Configuration Password
- SSL
- Generated 3DES Keys
- SSL & 3DES Keys Protection Summary
- Sun ONE Message Queue Access Controls
- Directory Credentials
- Persistent Storage Protection Summary

This security approach aims to prevent the following events from taking place:

- An eavesdropper intercepting a clear text password over the network
- An attacker manipulating a connector to change a user's password to a value of their choosing, which is equivalent to capturing the user's clear text password
- An attacker gaining access to a privileged component of Identity Synchronization for Windows
- An unprivileged user recovering a password from a file stored on disk.
- An intruder recovering a password from a hard disk that was removed from one of the components of the system. This could be a password being synchronized, or it could be a system password that is used to access a directory.

Configuration Password

To protect sensitive information while it is stored in the product's configuration registry and while it is transferred over the network, Identity Synchronization for Windows uses a configuration password. This configuration password is chosen by the administrator during core installation and is provided whenever the console or installer is run.

NOTE The System Manager must access the configuration password before passing it to the connector, therefore it stores it in its initialization file. File system access controls prevent access by non-privileged users to the System Manager's initialization file. The installer does not enforce any password policy for this password, see Hardening Your Security to increase security when selecting a configuration password.

SSL

Identity Synchronization for Windows can be configured to use LDAP over SSL everywhere that components use LDAP, the one exception to this is the installer. The installer cannot communicate with the Directory Server where the product's configuration registry is stored, and with the directories being synchronized over SSL. All access to the Sun ONE Message Queue is protected with SSL. SSL is mandatory between the Active Directory connector and Active Directory when synchronizing from the Sun ONE Directory Server to Active Directory.

NOTE When the synchronization settings are configured from Sun to Windows SSL is mandatory between the Active Directory connector and Active Directory. During connector installation, in order to search for a CA Certificate to store in the local AD connector certificate database the installer prompts you for Active Directory Credentials, the installer transfers this password in the clear as there is no SSL support in the installer. See the Hardening Your Security to avoid this security risk.

Generated 3DES Keys

A 3DES key generated from the configuration password is used to secure all sensitive information in the product's configuration registry. With the exception of log messages, all messages to the Message Queue are encrypted with per-topic 3DES keys. Messages sent between connectors and subcomponents are encrypted with per session 3DES keys. The Directory Server (subcomponent) plugin encrypts all user password changes with a 3DES key

SSL & 3DES Keys Protection Summary

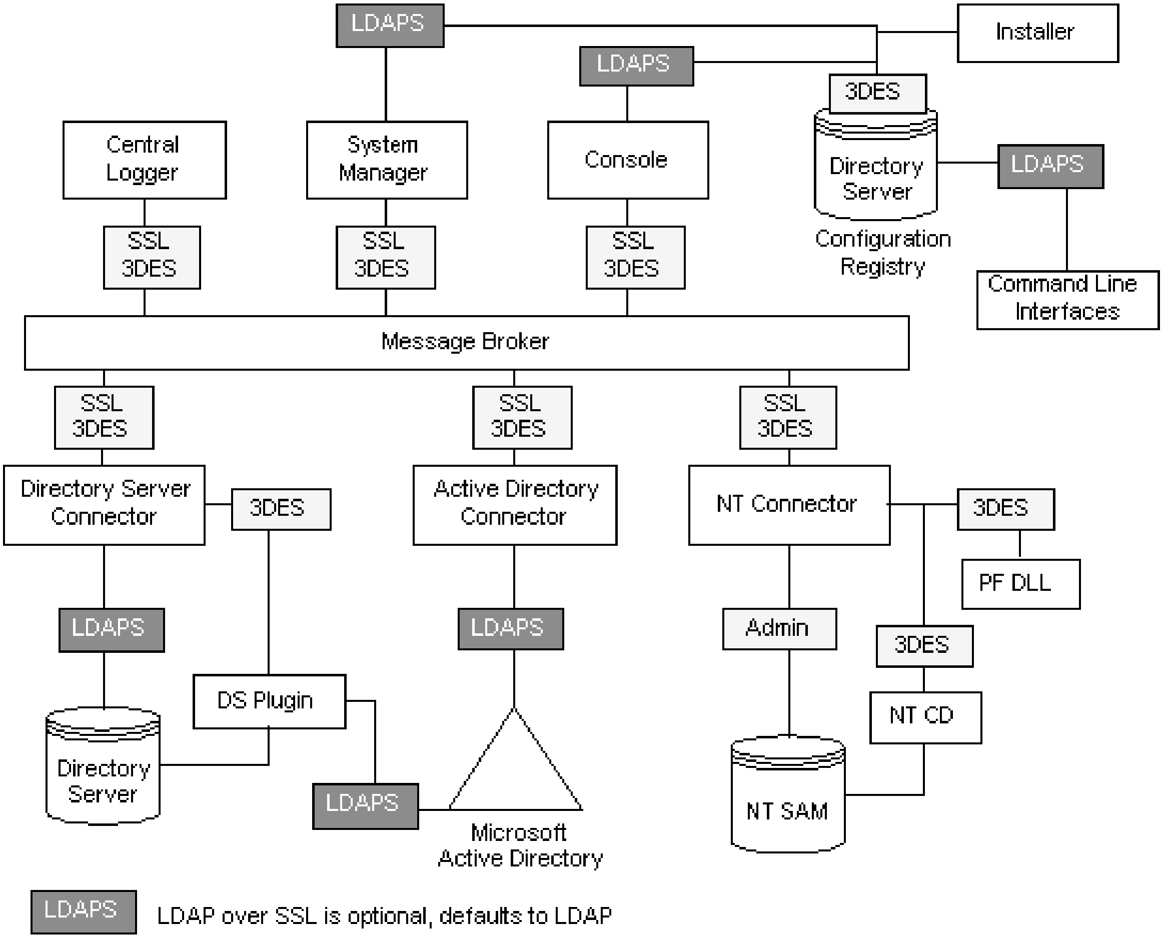
The following table summarizes how Identity Synchronization for Windows protects sensitive information that is sent over the network.

Table 10-1 Network Security

Optional LDAP over SSL between?	• The Directory Server connector and the Directory Server, The Active Directory connector and Active Directory, • The Directory Server plugin and Active Directory, • The command line interfaces and the product's configuration registry, • The console and the product's configuration registry, • The console and the Active Directory Global Catalog • The console and the Active Directory domains or the Directory Servers being synchronized • The Sun ONE Message Queue Broker and the product's configuration registry • The connectors, system manager, central logger, command line interface, and console may authenticate over LDAPS the Sun ONE Message Queue
Encrypted with 3DES keys	• All data between the DS connector and the Directory Server plugin • All data between the NT Connector and the NT Password Filter DLL, and the NT Change Detector • All sensitive information in the product's configuration registry. • All messages sent between connectors and subcomponents are encrypted with per-session 3DES keys • All messages sent over the Message Queue

Figure 10-1 contains an overview of the security features discussed in this section.

Figure 10-1 Identity Synchronization for Windows Security Overview



Sun ONE Message Queue Access Controls

Identity Synchronization for Windows uses Sun ONE Message Queue's access control to prevent unauthorized access to message subscription and publishing, allowing each connector to trust messages that it receives.

Unique username and passwords only known to Sun ONE Message Queue and to the connector are provided to access the Sun ONE Message Queue broker. Each message sent over the Sun ONE MQ is encrypted with a per topic 3DES key, which protects the message contents and prevents outsiders who do not know the topic key from sending meaningful messages. These measures prevent (a) an attacker from sending falsified password synchronization messages to connectors and (b) an attacker from impersonating a connector and receiving actual password updates.

NOTE By default clients of the Message Queue, such as the connectors and system manager, accept any SSL certificate that the Sun ONE Message Queue broker returns. See Hardening your Security for more information to enhance Message Queue certificate validation and other Message Queue-related security issues.

Directory Credentials

Privileged credentials are required by the connectors to change passwords in Active Directory and the Directory Servers being synchronized. These privileged credentials are encrypted before they are stored in the product's configuration registry.

Persistent Storage Protection Summary

The following tables summarize how Identity Synchronization for Windows protects sensitive information that is stored on disk.

Table 10-2 Persistent Storage Protection

Persistent Storage	Confidential Information	Protection
Product's configuration registry stored in a Directory Server	Credentials for accessing the directories and per Sun ONE Message Queue topic 3DES keys are stored in the product's configuration registry.	All sensitive information stored in the product's configuration registry is encrypted with a 3DES key that is generated from the configuration password. See Hardening your Security for recommendations to further protect the product's configuration registry.
Directory Server Retro Changelog	The Directory Server (subcomponent) plugin captures password changes and writes them to the Directory Server Retro Changelog.	The Directory Server (subcomponent) plugin encrypts all user password changes with a 3DES key per deployment.
Message Queue Broker Persistent Storage	The Message Queue Broker stores password synchronization messages sent between all connectors.	With the exception of log messages, all persisted messages are encrypted with per-topic 3DES keys.
Message Queue Broker Directory Credentials	The Message Queue Broker authenticates users against the product's configuration registry. It connects to the configuration registry using the directory administrative user name and password provided during core installation.	The directory password is stored in a passfile, which is protected with file system access controls.
System Manager Boot File	The system manager's boot file contains information for accessing the configuration. This includes the configuration password and the directory administrative user name and password provided during core installation.	This file is protected with file system access controls.

Persistent Storage	Confidential Information	Protection
Connectors and Central Logger Boot Files	Each connector as well as the central logger have an initial configuration file with credentials for accessing the Sun ONE Message Queue.	These files are protected with file system access controls.
Directory Server (subcomponent) plugin Boot Configuration	The plugin's configuration, stored under cn=config, includes credentials for connecting to the connector.	The cn=config subtree is protected with ACI's and the dse.ldif file, which mirrors this tree, is protected with file system access controls.
NT Password Filter DLL and NT Change Detector Boot Configuration	The NT subcomponent's configuration, which is stored in the Windows registry, includes credentials for connecting to the connector.	If access to the PDC's registry is not secure, these registry keys can be protected with access controls.
Windows connector's object cache	Windows connectors store hashed user passwords in the connector's object cache.	The passwords are not stored in the clear but encrypted with MD5 hashes (See Hardening you Security)

NOTE Active Directory and NT connectors object cache file permissions, which store unsalted MD5 hashes of user passwords, are set by default to 744. See Hardening your Security for more information to protect these stores from intruders.

Hardening your Security

This section depicts potential security weaknesses in the current release of the product and recommendations as to how to extend and harden security outside the product's default configuration. It includes the following:

- Configuration Password
- Active Directory Credentials during Installation
- Directory Server Administrative Credentials During Installation
- Directory Server User Credentials During Installation

- Product’s Configuration Registry Credentials During Installation
- Creating New product’s Configuration Registry Credentials
- Message Queue Client Certificate Validation
- Access to the Message Queue Broker
- Product’s Configuration Registry Certificate Validation
- User Passwords in the Windows connector’s Object Cache
- Restricting Access to the Product’s Configuration Registry

Configuration Password

The configuration password is used to protect sensitive configuration information but the installer does not enforce any password policy for this password; make sure that this password follows some strict guidelines choose a complex password that is not easily guessed and follow standard policy guidelines for strong passwords.

For example, it should be at least eight characters long, include upper case letters, lower case letters, and non-alphanumeric characters. It should not include your name, initials, or dates.

Active Directory Credentials during Installation

When the synchronization settings are configured from Sun to Windows SSL is mandatory between the Active Directory connector and Active Directory. During installation of the Active Directory Connector, in order to search for a CA Certificate, (this certificate is eventually stored in the local AD connector certificate database) the installer prompts you for Active Directory Credentials. Please note that while the installer provides “Administrator” as the default user in the installation panel, Administrator rights are not required to retrieve the certificate.

The installer cannot communicate with Active Directory over SSL. Therefore these Active Directory credentials, although not required to be administrative are transferred over the network in the clear.

Consequently, one of the following might be desirable,

- Retrieve the certificate (see Enabling SSL in the Active Directory Connector) and manually add it to the Active Directory connector’s certificate database.

- Create a temporary account used only for the purpose of reading the CA certificate, after which is deleted.
- Please note that the credentials do not need to be the Administrative.

Directory Server Administrative Credentials During Installation

The installer prompts you for administrative credentials for the Directory Server being synchronized when,

- Installing the Directory Server (subcomponent) plugin
- Installing the Directory Server Connector to generate the connector's user password

The installer cannot communicate with the Directory Sever over SSL; therefore, these credentials are transferred in the clear.

This is not an issue when installing the Directory Server (subcomponent) plugin as the installer must be run in the same machine where the Directory Server being synchronized is located; however, this is an issue when installing the Directory Server Connector as it does not need to be installed in the same machine as the Directory Server being synchronized. In the latter case one of the following might be desirable,

- Make sure that the network is secure between the machine where the connector is being installed and the Directory Server being synchronized
- Create a temporary administrative password in the Directory Server being synchronized only for the purpose of installing the Directory Server connector.
- Install the Directory Server Connector on the same machine as the Directory Server.

Directory Server User Credentials During Installation

When installing the Directory Server Connector the installer generates a Connector User password and is transferred to the Directory Server over the network in the clear.

- Make sure that the network is secure between the machine where the connector is being installed and the Directory Server being synchronized
- Install the Directory Server Connector on the same machine as the Directory Server.

Product’s Configuration Registry Credentials During Installation

Every time you run the installer it prompts you for the Directory Server location and credentials where the product’s configuration registry is stored. The installer cannot communicate with this Directory Server over SSL; therefore these credentials, which must be in the group Configuration Administrators (i.e.: admin), are not transferred over the network in the clear.

This is not an issue if core or any other component are installed on the same machine as the Directory Server used to store the products configuration registry.

- Make sure that the network is secure between the machine where the connector is being installed and the Directory Server that stores the product’s configuration registry.
- Install the product’s configuration registry in a different Directory Server than where the data being synchronized resides, this accomplishes reducing the scope of access to a Directory Server that does not also hold user data.
- Create new credentials in the Directory Server storing the product’s configuration registry only for the purpose of creating, modifying and updating the configuration registry; see Creating New product’s Configuration Registry Credentials for details.

Creating New product’s Configuration Registry Credentials

Credentials to access the Directory Server where the product’s configuration registry resides must be in the group Configuration Administrators. You might have a reason to create new credentials other than Admin, if so consider the following:

- The installer requires the user supply credentials for a user stored in the Console administrative subtree:

`ou=Administrators, ou=TopologyManagement, o=NetscapeRoot.`

- The core installer will not expand other users than “admin” into “uid=admin,ou=Administrators, ou=TopologyManagement, o=NetscapeRoot"). If you want to supply any username besides admin, you must specify the entire DN during core installation when you are first prompted for these credentials.
- Create the new user under ou=Administrators, ou=TopologyManagement, o=NetscapeRoot.
- Make sure the new credentials are in the group Configuration Administrators.
- Set ACI’s to allow only this user or perhaps all users in the group Configuration Administrators to access the Directory Server where the product’s configuration registry is stored. Documentation on managing access controls in the Sun ONE Directory Server can be found in the *Directory Server 5.2 Administrator’s Guide* Chapter 6 Managing Access Control.

Message Queue Client Certificate Validation

By default clients of the Message Queue, such as the connectors and system manager, accept any SSL certificate that the Sun ONE Message Queue broker returns.

1. To override this setting and force Message Queue clients to validate the Message Queue Broker’s certificate, edit:

```
<installation-root>/isw-<hostname>/resources/WatchList.properties,
```

2. Add the following to the JVM arguments of each process:

```
-Djavax.net.ssl.trustStore=<keystore path>
-DimqSSLIsHostTrusted=false
```

3. Restart the Identity Synchronization daemon or service.

The `javax.net.ssl.trustStore` property should point to a JSEE keystore that trusts the broker certificate, for example, `/etc/imq/keystore` can be used on the machine where core was installed because this is the same keystore used by the broker.

Message Queue Self-signed SSL certificate

By default the Sun ONE Message Queue broker uses a self-signed SSL certificate.

To install a different certificate, use the `keytool` utility that ships with Java to modify the broker's keystore (`/etc/imq/keystore` on Solaris and `<installation-root>/isw-<host name>/imq/etc/keystore` on Windows 2000). The alias of the certificate must be imq.

Access to the Message Queue Broker

By default the Sun ONE Message Queue uses dynamic ports for all services except for its port mapper. To access the broker trough a firewall or restrict the set of hosts that can connect to the broker, the broker should use fixed ports for all services.

This can be achieved by setting the imq.<service_name>.<protocol_type>.port broker configuration properties. Refer to the Sun ONE Message Queue Administrator's Guide for details.

Product's Configuration Registry Certificate Validation

The system manager accepts any certificate when connecting to the product's configuration registry over SSL; the Message Queue Broker accepts any certificate when connecting to the product's configuration registry over SSL. Currently, there is no way to make either the system manager or the Message Queue Broker validate the product's configuration registry SSL certificates.

User Passwords in the Windows connector's Object Cache

Active Directory and NT connectors store unsalted MD5 hashes of user passwords in the connector's object cache. In order to ensure maximum protection we recommend that you override default permissions of persist/ADP[nnn]/oc:

- In Windows by restricting access permissions to System and the Administrator's Group.
- In Solaris by changing the directory permissions for persist/ADP[nnn]/oc/ to 700.

Restricting Access to the Product’s Configuration Registry

When core is installed, the process of adding information to the Directory Server where the product’s configuration registry is stored does not include adding any access control information. To restrict access to only Configuration Administrators, the following ACI can be used:

```
(targetattr = "*") (target =
"ldap:///ou=IdentitySynchronization,ou=Services,dc=example,dc=com")
(version 3.0;acl "Test";deny (all)(groupdn !=
"ldap:///cn=Configuration Administrators, ou=Groups,
ou=TopologyManagement, o=NetscapeRoot");)
```

Documentation on managing access controls in the Sun ONE Directory Server can be found in the *Directory Server 5.2 Administrator’s Guide* Chapter 6 Managing Access Control.

Securing Replicated Configurations

Deployments connecting to Directory Servers using replication follow the same rules identified in Security Overview. This section gives an example replicated configuration and explains how to enable use of SSL in this configuration.

NOTE For an overview of planning, deploying, and securing replicated configurations see Installation Notes for Replicated Environments.

Table 10-3 lists the configuration components requiring CA certificates and identifies which certificates are required where.

Table 10-3 MMR Configuration Components Requiring CA Certificates

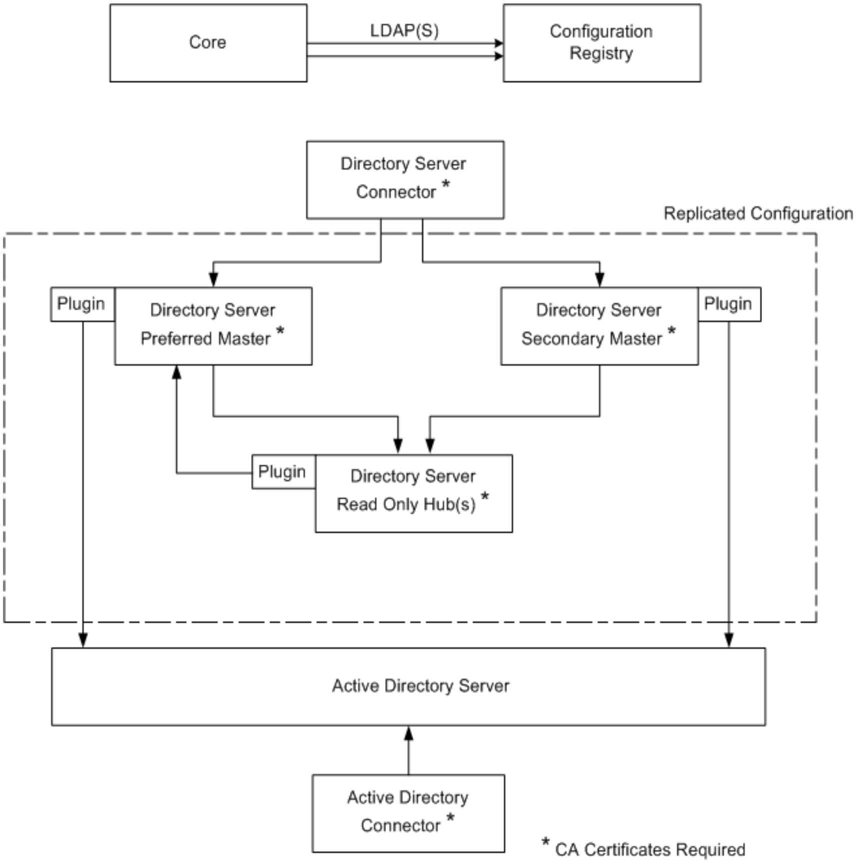
Component	Required CA certificates
Preferred Directory Server Replicated master	Active Directory system
Secondary Directory Server Replicated master	Active Directory system
Read-only Directory Server hub(s)	Preferred Directory Server replicated master Secondary Directory Server Replicated master
Directory Server connector	Preferred Directory Server replicated master Secondary Directory Server Replicated master
Active Directory connector	Active Directory system

Figure 10-2 shows Identity Synchronization for Windows installed in an MMR configuration. There are two replicated Directory Server masters with multiple Directory Server read-only hubs or consumers. Each Directory Server has its own plugin and there is only one Directory Server connector. There is one Active Directory system and one Active Directory connector.

NOTE When the Directory Server source is configured for SSL, you must make sure that both the preferred and secondary Directory Server certificates are trusted by the replica Directory Server.This is true for every Directory Server plugin of type `other` that you install on a system with a Directory Server hub or read-only replica.

Directory Server Plugin has access to the same CA certificates as its associated Directory server.

Figure 10-2 Replicated Configuration


Core
LDAP(S)
Configuration Registry
Directory Server Connector *
Replicated Configuration
Plugin
Directory Server Preferred Master *
Directory Server Secondary Master *
Plugin
Plugin
Directory Server Read Only Hub(s) *
Active Directory Server
Active Directory Connector *
* CA Certificates Required

Using idsync certinfo

Use the `idsync certinfo` utility to determine what certificates are required based on the current Identity Synchronization for Windows SSL settings. Execute `idsync certinfo` to retrieve information about what certificates are required in each certificate database.

NOTE You must be sure that when you are configuring the Directory Server source for SSL, both the preferred and secondary Directory Server source certificates are trusted by the replica Directory Server for all Directory subcomponents or plugins.

Arguments

`idsync certinfo` accepts the following arguments:

Argument	Description
-h	Configuration registry hostname
-p	Configuration registry LDAP port number
-D	Configuration registry bind DN
-w	Configuration registry bind password
-s	Configuration registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password

Usage

The following example uses `idsync certinfo` to search for system components designated to run under SSL communications. The results of this example identifies two connectors (CNN101 and CNN100) and provides instructions as to where to import the appropriate CA certificate.

```
C:\Program Files\Sun\MPS\isw-hostname\bin> idsync certinfo -h hostname -p 388 -D
"cn=Directory Manager" -w dirmanager -s dc=example,dc=com -q <password>

Connector: CNN101
```

```
 Certificate Database Location: C:\Program Files\Sun\MPS\isw-hostname\etc\CNN101
 Certificate Database Password:   LTD8a4kfXtMe95CX
 Connector Server Certificate Name:
 Get 'Active Directory CA' certificate from Active Directory and import into
 Active Directory Connector certificate db for server
 ldaps:://hostname.example.com:636

Connector: CNN100

 Certificate Database Location:   C:\Program
 Files\Sun\MPS\isw-hostname\etc\CNN100
 Certificate Database Password:   d1JESIKFRfriCFnq
 Connector Server Certificate Name:   server-cert100
 Export 'Directory Server CA' certificate from Directory Server certificate db
 and import into Directory Server Connector certificate db
 ldaps://hostname.example.com:636
 Export 'Active Directory CA' certificate from Active Directory Server
 hostname.example.sun.com:389 and import into Directory Server Server certificate
 db for server ldaps://hostname.example.com:638

SUCCESS
```

NOTE Use the `certutil` version bundled with Directory Server 5.2. Later versions of `certutil` are incompatible with Identity Synchronization for Windows.

Enabling SSL in Directory Server

Follow these steps to enable SSL in a Directory Server.

NOTE These abbreviated procedures are for your convenience. Refer to the *Directory Server 5.2 Administrator's Guide* for more information.

NOTE While `certutil` is provided with the Directory Server in most cases, it is not included in the version of the Directory Server installed from Solaris packages. Therefore, if someone has installed their server using Solaris packages rather than the zip installation, they will need to get download the Directory Server Resource Kit (DSRK), which does contain `certutil` and other utilities for interacting with NSS certificate databases.

Use the `certutil` version bundled with Directory Server 5.2. Later versions of `certutil` are incompatible with Identity Synchronization for Windows.

1. Create a new key certificate database for the Directory Server by entering:

   ```
   C:\Program Files\Sun\MPS\shared\bin\certutil.exe -N -d . -P
   slapd-hostname
   ```

   ```
   In order to finish creating your database, you
   must enter a password which will be used to
   encrypt this key and any future keys.
   The password must be at least 8 characters long,
   and must contain at least one non-alphabetic character.
   Enter new password:
   Re-enter password:
   ```

NOTE These examples are run in the `alias` directory immediately below the server root. Otherwise, the Directory Server will not be able to find the certificate database.

2. Generate a self-signed certificate - this will be the server certificate used by the Directory Server. Make sure you choose the subject DN according to the hostname of the server where the Directory Server is running.

NOTE By default, a self-signed certificate is valid for three months. If you want to increase or decrease this time period, use the `-v` <*`months-valid`*> option. For example, to increase the time period to 24 months, enter `-v 21` or to decrease the time period to one month, enter `-v -2`.

```
C:\Program Files\Sun\MPS\shared\bin\certutil.exe -d . -P
slapd-hostname -S -n server-cert -s
"cn=hostname.example.com,c=us" -x -t CTu,,
```

```
A random seed must be generated that will be used in the
creation of your key.  One of the easiest ways to create a
random seed is to use the timing of keystrokes on a keyboard.
To begin, type keys on the keyboard until this progress meter
is full.  DO NOT USE THE AUTOREPEAT FUNCTION ON YOUR KEYBOARD!
Continue typing until the progress meter is full:
|************************************************************|
Finished.  Press enter to continue:
Enter Password or Pin for "NSS Certificate DB":
Generating key.  This may take a few moments...
```

3. Display the certificates for checking purposes.

```
C:\Program Files\Sun\MPS\shared\bin\certutil.exe -L -d . -P
slapd-hostname
```

```
Certificate Name       Trust Attributes

server-cert            CTu,,

p    Valid peer
P    Trusted peer (implies p)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

4. Create a PIN file, so that certificate database password does not have to be entered each time the Directory Server is restarted.

```
C:\Program Files\Sun\MPS\alias > echo Internal (Software)
Token:secret12 > slapd-hostname-pin.txt
```

5. Enable SSL in the Directory Server:

   a. Open the console.

   b. Select the Configuration tab.

   c. Select the Encryption tab (on the right pane).

   d. Check Enable SSL for this server.

   e. Check Use this cipher family: RSA.

   f. Click on save and on OK twice.

   g. Select the Network tab.

h. Update the Secure Port field. If running on the same machine as Active Directory, the port must be changed from 636 to an unused port or the Directory Server will not start.

i. Click on Save, then yes, then OK.

j. Select the Tasks tab (on the top).

k. Click on Restart Directory Server, then click on yes.

Retrieving the CA Certificate from the Directory Server Certificate Database

Ensure that you have enabled SSL in the Directory Server. To export the Directory Server certificate to a temporary file so that you can import it into the certificate database of the Directory Server connector, issue the following command:

```
C:\Program Files\Sun\Sun\MPS\shared\bin\certutil.exe -L -d . -P
slapd-hostname -n server-cert -a > C:\s-cert.txt
```

These examples are run in the alias directory immediately below the server root. Otherwise, the Directory Server will not be able to find the certificate database.

Enabling SSL in the Active Directory Connector

Enabling SSL in the Active Directory takes several steps:

- Retrieving the Active Directory CA Certificate
- Adding the Active Directory Certificate to the Connector’s Certificate Database

Retrieving the Active Directory CA Certificate

NOTE The Active Directory CA certificate can be retrieved in two ways:

Retrieving the Active Directory CA Certificate using certutil

From the Active Directory machine run the following command to export the certificate. Note that certutil is a program that ships with Windows 2000.

```
C:\>certutil -ca.cert cacert.bin
```

The cacert.bin file can then be imported into a certificate database.

Retrieving the Active Directory CA Certificate over LDAP

1. Execute the following search against Active Directory

```
ldapsearch -h hostname -D administrator DN -w administrator password -b
"cn=configuration,dc=put,dc=your,dc=domain,dc=here" "cacertificate=*"
```

   where administrator DN looks like `cn=administrator,cn=users,dc=put,dc=your,dc=domain,dc=here` (the domain name in this case is: `put.your.domain.name.here`)

   There will be several entries matching the above search filter. You probably need the one that has `cn=Certification Authorities, cn=Public Key Services` in its DN.

2. Open up a text editor and cut out the first value of the first CA certificate attribute (it should be a BASE-64 encoded text block). Paste those lines into the text editor (only the value). Edit the contents, so that none of the lines start with white space.

3. Add `-----BEGIN CERTIFICATE-----` before the first line and `-----END CERTIFICATE-----` after the last line. So, in the end you should have something like this:

```
-----BEGIN CERTIFICATE-----

MIIDvjCCA2igAwIBAgIQDgoyk+Tu14NGoQnxhmNHLjANBgkqhkiG9w0BAQUFA
DCBjjEeMBwGCSqGSIb3DQEJARYPYmVydG9sZEBzdW4uY29tMQswCQYDVQQGEwJVUzELMAkGA1UEC
BMCVFgxDzANBgNVBAcTBkF1c3RpbjEZMBcGA1UEChMQU3VuIE1pY3Jvc3lzdGVtczEQMA4GA1UEC
xMHaVBsYW5ldDEUMBIGA1UEAxMLUmVzdGF1cmFudHMwHhcNMDIwMTExMDA1NDA5WhcNMTIwMTExM
DA1OTQ2WjCBjjEeMBwGCSqGSIb3DQEJARYPYmVydG9sZEBzdW4uY29tMQswCQYDVQQGEwJVUzELM
AkGA1UECBMCVFgxDzANBgNVBAcTBkF1c3RpbjEZMBcGA1UEChMQU3VuIE1pY3Jvc3lzdGVtczEQM
A4GA1UECxMHaVBsYW5ldDEUMBIGA1UEAxMLUmVzdGF1cmFudHMwXDANBgkqhkiG9w0BAQEFAANLA
DBIAkEAyekZa8gwwhw3rLK3eV/12St1DVUsg31LOu3CnB8cMHQZXlgiUgtQ0hm2kpZ4nEhwCAHhF
LD3iIhIP4BGWQFjcwIDAQABo4IBnjCCAZowEwYJKwYBBAGCNxQCBAYeBABDAEEwCwYDVR0PBAQDA
gFGMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYEFJ5Bgt6Oypq7T8Oykw4LH6ws2d/IMIIBMgYDV
R0fBIIBKTCCASUwgdOggdCggc2GgcpsZGFwOi8vL0NOPVJlc3RhdXJhbnRzLENOPWRvd2l0Y2hlc
ixDTj1DRFAsQ049UHVibGljJTIwS2V5JTIwU2VydmljZXMsQ049U2VydmljZXMsQ049Q29uZmlnd
XJhdGlvbixEQz1yZXN0YXVyYW50cyxEQz1jZW50cmFsLERDPXN1bixEQz1jb20/Y2VydGlmaWNhd
GVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdGNsYXNzPWNSTERpc3RyaWJ1dGlvblBvaW50ME2gS
6BJhkdodHRwOi8vZG93aXRjaGVyLnJlc3RhdXJhbnRzLmNlbnRyYWwuc3VuLmNvbS9DZXJ0RW5yb
2xsL1Jlc3RhdXJhbnRzLmNybDAQBgkrBgEEAYI3FQEEAwIBADANBgkqhkiG9w0BAQUFAANBAL5R9
R+ONDdVHWu/5Sd9Tn9dpxN8oegjS88ztv1HD6XSTDzGTuaaVebSZV3I+ghSInsgQbH0gW4fGRwaI
BvePI4=
-----END CERTIFICATE-----
```

4. Save this into a file, e.g. ad-cert.txt.

   The ad-cert.txt file can then be imported into a certificate database.

Adding the Active Directory Certificate to the Connector's Certificate Database

These steps only need to be followed if SSL is enabled for the Active Directory connector after the connector was installed or if invalid credentials were provided during installation.

1. On the machine where the Active Directory connector is installed, stop the Identity Synchronization for Windows service/daemon.
2. Retrieve the Active Directory CA certificate and save it to a file called `cacert.bin`.

NOTE This procedure assumes that the CA certificate was obtained using the Active Directory `certutil` tool (see Retrieving the Active Directory CA Certificate using certutil) and will not work if it was obtained over LDAP in ASCII form.

3. Assuming the Active Directory connector has connector ID CNN101 (see logs/central/error.log for a mapping from connector ID to the directory source it manages), go to its certificate database directory on the machine where it was installed, and import the cacert.bin file:

```
C:\\Program Files\Sun\Sun\MPS\isw-hostname\shared\bin\certutil.exe -A -d . -n
ad-ca-cert -t C,, -i \cacert.bin
```

NOTE If the certificate was obtained in ASCII form over LDAP, then you should add a "-a" argument to the `certutil` command line to indicate that it is in ASCII form rather than binary.

4. Restart the Identity Synchronization for Windows service/daemon.

NOTE Because the certutil.exe is installed automatically when you install Directory Server 5.2, you will not be able to add a CA certificate to a connector when the connector is installed on another host.

At a minimum, you must install the Sun ONE Server Basic Libraries and Sun ONE Server Basic System Libraries from the Directory Server 5.2 package on the server where the Active Directory connector is installed. (You do not have to install the Administration Server or Directory Server components.)

In addition, be sure to select the JRE subcomponent from the console (to ensure your ability to uninstall).

Adding the Active Directory Certificate to the Directory Server

Follow these steps to add the Active Directory CA certificate to the Directory Server certificate database.

NOTE Make sure that you have enabled SSL in the Directory Server.

1. Retrieve the Active Directory CA certificate and save it to a file called cacert.bin.
2. Stop the Directory Server.
3. On the machine where Directory Server is installed, import the Active Directory CA certificate:

   ```
   C:\Program Files\Sun\MPS\shared\bin\certutil.exe -A -d . -P
   slapd-hostname- -n ad-ca-cert -t C,, -i cacert.bin
   ```

NOTE If the certificate was obtained in ASCII form over LDAP, then you should add a "-a" argument to the `certutil` command line to indicate that it is in ASCII form rather than binary.

4. Start the Directory Server.

Adding the Directory Server Certificate to the Directory Server Connector

1. On the machine where the Directory Server connector is installed, stop the Identity Synchronization for Windows service/daemon.
2. Retrieve the Directory Server CA certificate.
3. Assuming the Directory Server connector has connector ID CNN100 (see logs/example/error.log for a mapping from connector ID to the directory source it manages), go to its certificate database directory on the machine where it was installed, and import the cacert.bin file:

   ```
   C:\Program Files\Sun\MPS\shared\bin\certutil.exe -A -d . -n
   ds-cert -t C,, -i C:\s-cert.txt
   ```

NOTE If the certificate was obtained in ASCII form over LDAP, then you should add a “-a” argument to the `certutil` command line to indicate that it is in ASCII form rather than binary.

4. Restart the Identity Synchronization for Windows service / daemon.

Part 1

Appendices

Appendix A

Command Line Utilities

Identity Synchronization for Windows includes a number of command line utilities that are useful in installing the product. This Appendix explains command line usage and lists these utilities in the following sections:

- Using idsync
- Using changepw
- Using importcnf
- Using startsync
- Using stopsync
- Using certinfo
- Using printstat
- Using resetconn

NOTE	Several of these scripts are necessary for successful installation and are explained in the section where they are used. Other optional scripts are listed in this Appendix.

Using idsync

All Identity Synchronization for Windows command line utilities are executed with the **idsync** script. Use the **idsync** script as follows:

```
idsync <subcommand> <subcommand-arg1> <subcommand-arg2> ...
```

The first argument is a sub command followed by its arguments. Identity Synchronization for Windows sub commands include:

- **importcnf.** The import configuration script allows you to do core configuration via the configuration XML document. This document contains enough information to configure an Identity Synchronization for Windows installation. See "Using importcnf" on page 187 for more information.
- **changepw**. Changes passwords. See "Using changepw" on page 186 for more information.
- **prepds.** A script that prepares the Sun Directory Server for installation. See "Prepare Directory Server" on page 86 for more information.
- **linkusers.** Links existing users as part of the installation. See "Linking Users" on page 106 for more information.
- r**esync.** Resynchronizes existing users as part of the installation. See "User Resynchronization" on page 111 for more information.
- **startsync.** Starts synchronization from the command line as opposed to using the console. See "Using startsync" on page 199 for more information.
- **stopsync.** Stops synchronization from the command line as opposed to using the console. See "Using stopsync" on page 200 for more information.
- **certinfo.** Displays certificate information. See "Using certinfo" on page 201 for more information.
- **printstat.** Prints configuration status. See "Using printstat" on page 203 for more information.
- **resetconn**. Resets connector state in the Configuration Registry to uninstalled. See "Using resetconn" on page 205 for more information.

NOTE When you type the idsync command in a terminal window, any DN-valued arguments, such as the manager bind DN or suffix name, are converted from the character set of that window to UTF-8 before being sent to the directory server. The use of backslashes as escape characters is not permitted in suffix names.

If you wish to specify UTF-8 characters on Solaris, you must ensure that your terminal window has a locale which is based on UTF-8. Type 'setenv' and check the environment variable LC_CTYPE, or if not set, LANG.

Entering Passwords

Anywhere that a <*bind password*> or <*configuration password*> value is required, the value "-" can be used to read the password from STDIN or console. When "-" is the value of multiple password options, the idsync program will prompt for passwords in the order of the arguments.

Accessing the Configuration Directory Server Via SSL

All of the sub commands list the following options, which provide information about securely accessing the configuration Directory Server via SSL:

```
[-Z] [-P <cert db path>] [-m <secmod db path>]
```

Getting Help

Enter any of the following three help options following a sub command:

```
-help --help -?
```

The usage for the sub command is listed to the command console. If no sub command is specified, usage for idsync is listed, including a list of sub commands.

Using changepw

The script changepw changes the configuration password for Identity Synchronization for Windows. Before running the script, stop the console, any running installers, and all other system processes. Export the ou=Services tree in the Configuration Registry to ldif as backup.

Arguments

The script changepw uses the following arguments:

Table A-1 changepw Arguments

Argument	**Description**
-h	Configuration registry hostname
-p	Configuration registry port no
-D	bind distinguished name
-w	bind password
-s	This is the name of the configuration registry rootsuffix. A rootsuffix is a distinguished name such as dc=example,dc=com.
-q	configuration password
-b	new configuration password

Usage

To change the configuration password for Identity Synchronization for Windows:

1. Stop all of the Identity Synchronization processes (for example, System Manager, Central Logger, Connectors, Console, Installers / Uninstallers)
2. After stopping all the processes, make a backup of the ou=Services tree by exporting the configuration registry to ldif.
3. Enter the following command.

```
idsync changepw -h hostname -p <port no> -D <bind DN> -w <bind
password> -s <rootsuffix> -q <configuration password> -b <new
configuration>
```

```
Are you sure that want to change the configuration password
(y/n)? yes
```

```
Before restarting the system - you must edit the
$PSWHOME/resources/SystemManagerBootParams.cfg file and change
the 'deploymentPassword' to the new value.
```

```
SUCCESS
```

4. Modify the `SystemManagerBootParams.cfg` file that appears in `$PSWHOME\resources` before restarting the system. This file contains the configuration password that the SystemManager uses to connect to the configuration registry. For example a password `'oldpassword'` - change the line:

   ```
   <Parameter name="manager.configReg.deploymentPassword"
   value="oldpassword"/>
   ```

   to:

   ```
   <Parameter name="manager.configReg.deploymentPassword"
   value="newpassword"/>
   ```

5. If there were any errors reported, restore the configuration registry using the ldif from Step 2 and then try again. The most likely reason for an error is that the Directory Server hosting the configuration registry goes down during the password change.

Using importcnf

The administrator has the alternative to provide configuration information via a command line interface as opposed to using the Console; this interface is called `idsync importcnf`. The directory deployment's topology can be documented in an XML Document that we shall name the Configuration XML Document. It contains enough information to configure a Identity Synchronization for Windows installation.

Run `idsync importcnf` after installing core. `idsync importcnf` can be run multiple times following core installation, but should not be run following the installation of connectors or connector sub-components.

Any error detected in the command line input configuration file results in a fatal error that will abort the command and provide necessary information to correct the mistakes. Possible errors can be caused by:

- Illegal XML in the Configuration XML Document contains e.g. missing closing element, missing closing string quote, unrecognized element etc.

- Omitted required XML elements or attributes e.g. SunDirectoryGlobals, userObjectClass etc.
- Illegal use of configuration elements e.g. mapping 2 different AttributeDescriptions from ActiveDirectoryGlobals to the same SunDirectoryGlobals AttributeDescription.
- Failure to connect to the configuration registry.

Arguments

`importcnf` accepts the following arguments:

Table A-2 `importcnf` Arguments

Argument	**Description**
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-n	[Validate the configuration file, but do not update the configuration registry.
-f <*filename*>	Names the Configuration XML Document input file

importcnf XML Document

The `importcnf` script uses as input an XML document that contains the same information that an administrator would provide via the Console’s UI, namely information about Directory Sources, Directory Globals, and Synchronization User Lists.

NOTE Complete, commented samples are provided in the `isw-`*`hostname`*`/samples` directory where you installed Identity Synchronization for Windows.

Configuration XML Document Usage Samples

Following are usage samples for defining Creation / Significant Attributes, Creation / Modification Synchronization Settings, and Attribute Maps in the Configuration XML Document used by `importcnf`.

Defining Directory Sources

Single Master Sun, Single AD Domain with Single Synchronized User List

This example assumes the synchronization settings have been set to synchronize modifications and creations from Windows to Sun.

The SunDirectorySource synchronized user list Synchronization Scope Definition Sets (SSDS) is either the Windows or the Sun Directory Source side of a Synchronization User List. A Sychronization User List will always have one SSDS for a Windows Directory Source and one SSDS for a Sun Directory Source.

SSDS definition includes a creation expression (to define the parent DN and naming attribute) for when new entries are propagated from AD to Sun.

NOTE The XML will not work with out the surrounding `ActiveConfiguration> ... </ActiveConfiguration>` tags.

```
<SunDirectorySource displayName="dc=example,dc=com">
  <SynchronizationHost hostname="ds-host.example.com" port="389"/>
  <SyncScopeDefinitionSet sulid="SUL1"
    location="ou=people,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=people,dc=example,dc=com"/>
</SunDirectorySource>
```

The AD synchronization host requires credentials, the Sun one does not.

```
<ActiveDirectorySource displayName="example.com">
  <SynchronizationHost hostname="ad-host.example.com">
    <Credentials userName="cn=Administrator,cn=users, dc=example,
       dc=com" cleartextPassword="examplePassword"/>
  </SynchronizationHost>
  <SyncScopeDefinitionSet sulid="SUL1"
```

```
    location="cn=Users,dc=example,dc=com"/>
</ActiveDirectorySource>
```

Single Master Sun, Single AD Domain, Single NT Domain with Multiple SULs

This example assumes the synchronization settings have been set to synchronize modifications and creations from Sun to Windows.

The `ActiveDirectorySource` synchronized user list SSDS definition includes a creation expression (to define the parent DN and naming attribute) for when new entries are propagated from Sun to AD; NT does not require creation expressions.

```
<SunDirectorySource displayName="dc=example,dc=com">
  <SynchronizationHost hostname="ds-host.example.com" port="389"/>
  <SyncScopeDefinitionSet sulid="SULAD"
    location="ou=people,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=people,dc=example,dc=com"
    index=1/>
  <SyncScopeDefinitionSet sulid="SULNT"
    location="ou=ntpeople,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=ntpeople,dc=example,dc=com"
    index=2/>
</SunDirectorySource>


<ActiveDirectorySource displayName="example.com">
  <SynchronizationHost hostname="ad-host.example.com">
    <Credentials userName="cn=Administrator,cn=users,dc=example,dc=com"
      cleartextPassword="examplePassword"/>
  </SynchronizationHost>
  <SyncScopeDefinitionSet sulid="SULAD" location="cn=Users,dc=example,dc=com"/>
</ActiveDirectorySource>


<NTDirectorySource displayName="ntexample1">
  <SynchronizationHost hostname="NTHOST1"/>
  <SyncScopeDefinitionSet sulid="SULNT"/>
```

```
</NTDirectorySource>
```

Single Master Sun, Single AD Domain, with Multiple SULs using filters

This example assumes the synchronization settings have been set to synchronize modifications and creations between Sun and Windows.

The ActiveDirectorySource and the SunDirectorySource synchronized user list SSDS definition both include creation expressions (to define the parent DN and naming attribute) for when new entries are propagated between Sun and AD.

```
<ActiveDirectorySource displayName="example.com">
  <SynchronizationHost hostname="ad-host.example.com">
  <Credentials userName="cn=Administrator,cn=users,dc=example,dc=com"
    cleartextPassword="examplePassword"/>
  </SynchronizationHost>
  <SyncScopeDefinitionSet sulid="SUL_SALES"
    location="cn=Users,dc=example,dc=com"
    filter="(department=sales)"
    index="1"/>
  <SyncScopeDefinitionSet sulid="SUL_FINANCE"
    location="cn=Users,dc=example,dc=com"
    filter="(department=finance)"
    index="2"/>
</ActiveDirectorySource>

<SunDirectorySource displayName="dc=example,dc=com">
  <SynchronizationHost hostname="ds-host.example.com" port="389"/>
  <SyncScopeDefinitionSet sulid="SUL_SALES"
    location="ou=sales,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=sales,dc=example,dc=com"
    index=1/>
  <SyncScopeDefinitionSet sulid="SUL_FINANCE"
    location="ou=finance,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=finance,dc=example,dc=com"
```

```
index=2/>
```

Defining SULs Using Filters and Indices

NOTE For more information about using filters and indices, see "Synchronization User List Definitions and Configuration" on page 213.

Following are two usage samples for using filters and indices.

Using filters to differentiate same Base DNs

Reversing the indices is not valid because the filter on SUL_MANAGER is more specific than the filter on SUL_PEOPLE.

```
<SunDirectorySource ...>
  ...
<SyncScopeDefinitionSet sulid="SUL_MANAGER"
    location="ou=people,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=people,dc=example,dc=com"
    filter="(employeeType=manager)"
    index=1/>
<SyncScopeDefinitionSet sulid="SUL_PEOPLE"
    location="ou=people,dc=example,dc=com"
    creationExpression="cn=%cn%,ou=people,dc=example,dc=com"
    index=2/>
</SunDirectorySource>
```

Defining Synchronization Settings, Significant/Creation Attributes, and Attribute Maps

Attribute flow settings, significant and creation attributes, and attribute maps are only specified once for each directory source type as Directory Globals.

Synchronizing Creates and Modifies from Sun to Windows

In a Sun/Active Directory deployment

When synchronizing creates from Sun to Active Directory, the Active Directory Source's mandatory creation attributes must be mapped to attributes in the Sun Directory Source; the maps are used during creation of a new User in Active Directory. Modifications are synchronized for `employeeNumber` only; note that `employeeNumber` will also be synchronized from Sun to Active Directory during creation.

```
<SunDirectoryGlobals
  FlowInboundCreates="TRUE"
  FlowInboundModifies="TRUE"
  FlowOutboundCreates="FALSE"
  FlowOutboundModifies="FALSE"
  userObjectClass="inetOrgPerson">
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="employeeNumber"/>
</SunDirectoryGlobals>


<ActiveDirectoryGlobals
  FlowInboundCreates="FALSE"
  FlowInboundModifies="FALSE"
  FlowOutboundCreates="TRUE"
  FlowOutboundModifies="TRUE">
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="employeeID"/>
```

The following maps are required from the you to satisfy Active Directory mandatory creation attributes:

```
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="cn"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="cn"/>
</AttributeMap>
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="uid"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="samaccountname"/>
</AttributeMap>
```

The following map is required to synchronize modifications to `employeeNumber`:

```
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="employeeNumber"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="employeeID"/>
</AttributeMap>
```

In a Sun/NT deployment

When synchronizing creates from Sun to NT this Directory Source’s mandatory creation attributes must be mapped to attributes in the Sun Directory Source; the maps are used during creation of a new User in NT. Modifications are synchronized for comments only; note that comments will also be synchronized from Sun to NT during creation.

```
<SunDirectoryGlobals
  FlowInboundCreates="TRUE"
  FlowInboundModifies="TRUE"
  FlowOutboundCreates="FALSE"
  FlowOutboundModifies="FALSE"
  userObjectClass="inetOrgPerson">
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="description"/>
</SunDirectoryGlobals>



<NTDirectoryGlobals
  FlowInboundCreates="FALSE"
  FlowInboundModifies="FALSE"
  FlowOutboundCreates="TRUE"
  FlowOutboundModifies="TRUE">
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="USER_COMMENT"/>
  <AttributeMap parent.attr="AttributeMap">
    <AttributeDescription parent.attr="SunAttribute"
                          name="uid"/>
    <AttributeDescription parent.attr="WindowsAttribute"
                          name="USER_NAME"/>
  </AttributeMap>
```

The following map is required to synchronize modifications to description.

The following map is required for the NT mandatory creation attribute:

```
  <AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="description"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="USER_COMMENT"/>
  </AttributeMap>
</NTDirectoryGlobals>
```

Synchronizing Creates from Windows to Sun

In these examples the Directory Server `userObjectclass` is `inetorgperson`; therefore, `cn` and `sn` must be defined as a creation attributes on the Directory Server side, and have maps on the Active Directory side. Modifications are synchronized for `telephonenumber` and `password` only; note that `telephonenumber` (by default) will also be synchronized from Active Directory to Directory Server during creation.

In a Sun/AD deployment

```
<SunDirectoryGlobals
  FlowInboundCreates="FALSE"
  FlowInboundModifies="FALSE"
  FlowOutboundCreates="TRUE"
  FlowOutboundModifies="TRUE"
  userObjectClass="inetOrgPerson">
```

The cn and sn creation attributes are required to successfully create user entries of type inetorgperson; cn is set as creation "only," while "sn" is set as a significant, which implies inclusion in "both" creates and modifies.

```
  <AttributeDescription parent.attr="CreationAttribute"
                        name="cn"/>
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="sn"/>
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="telephonenumber"/>
</SunDirectoryGlobals>
<ActiveDirectoryGlobals
  FlowInboundCreates="TRUE"
  FlowInboundModifies="TRUE"
  FlowOutboundCreates="FALSE"
  FlowOutboundModifies="FALSE">
  <AttributeDescription parent.attr="SignificantAttribute"
                        name="telephonenumber"/>
```

The following maps are required to match the Sun objectclass inetorgperson creation syntax:

```
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="cn"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="cn"/>
</AttributeMap>
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="sn"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="sn"/>
</AttributeMap>
```

The following map is required to synchronize modifications to `telephonenumber`:

```
<AttributeMap parent.attr="AttributeMap">
  <AttributeDescription parent.attr="SunAttribute"
                        name="telephonenumber"/>
  <AttributeDescription parent.attr="WindowsAttribute"
                        name="telephonenumber"/>
</AttributeMap>
</ActiveDirectoryGlobals>
```

The following map is required to synchronize modifications to description:

```
<AttributeMap parent.attr="AttributeMap">
<AttributeDescription parent.attr="SunAttribute"
name="description"/>
<AttributeDescription parent.attr="WindowsAttribute"
name="USER_COMMENT"/>
</AttributeMap>
</ActiveDirectoryGlobals>
```

Using startsync

Start synchronization from the command line.

Arguments

`startsync` accepts the following arguments:

Table A-3 `startsync` Arguments

Argument	Description
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.

Argument	Description
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Using stopsync

Stops synchronization from the command line.

Arguments

`stopsync` accepts the following arguments:

Table A-4 `stopsync` Arguments

Argument	Description
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.

Argument	Description
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Using certinfo

Displays certificate information, which includes the certificates that must be added to each connector or Directory Server plug-in certificate database.

Arguments

`certinfo` accepts the following arguments:

Table A-5 `certinfo` Arguments

Argument	Description
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN

Argument	Description
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Using printstat

Prints out the status of the installed connectors, the system manager, and the message queue. The status can be any of the following:

- **Uninstalled.** The connector has not been installed.
- **Installed.** The connector has been installed, but is not ready for synchronization yet. It has not received its runtime configuration yet.
- **Ready.** The connector is ready for synchronization, but it is currently not synchronizing any objects yet.
- **Syncing.** The connector is synchronizing objects.

Arguments

`printstat` accepts the following arguments:

Table A-6 `printstat` Arguments

Argument	Description
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.

Argument	Description
-P <cert db path>	Specify the path and filename of the client’s certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server’s own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Using resetconn

Resets the state of connectors in the Configuration Directory to *uninstalled.* `resetconn` provides two options:

- **-e** <*name of directory source*>: Specifies the name of the directory source to be reset. (Connectors are identified in the installers by their directory source name.)
- **-n** (safe mode): Indicates whether the arguments specified for the command are correct without doing any work.

Arguments

`resetconn` accepts the following arguments:

Table A-7 `resetconn` Arguments

Argument	**Description**
-h	Configuration Registry hostname
-p	Configuration Registry LDAP port no
-D	Configuration Registry bind DN
-w	Configuration Registry bind password
-s	Configuration Registry rootsuffix. A rootsuffix is a distinguished name such as `dc=example,dc=com`.
-q	Configuration password
-Z	Specify that SSL be used to provide certificate-based client authentication.
-N	Specify the certificate name to use for certificate-based client authentication, for example: `-N "Directory-Cert"`.

Argument	Description
-P <cert db path>	Specify the path and filename of the client's certificate database. This file may be the same as the certificate database for an SSL-enabled version of Directory Server. When using the command on the same host as the Directory Server you may use the server's own certificate database, for example: `-P <installDir>/alias/slapd-<serverId>-cert7.db` If `-Z` is specified and `-P` is not, the `<cert db path>` defaults to `<current working directory>/cert7.db`. Note: If the certificate database file is not found in the specified directory, an *empty* database will be created in that directory. The database consists of three files: `cert7.db`, `key3.db`, and `secmod.db`.
-m <secmod db path>	Specify the path to the security module database. For example: `/var/Sun/MPS/slapd-<serverID>/secmod.db` You need to specify this option only if the security module database is in a different directory from the certificate database itself.

Appendix B

LinkUsers XML Document Sample

To customize to your environment use the LinkUsers XML Document titled `IlodeLinkUsersIntegrate.cfg`. For this particular integration test, the canned LinkUsers XML Document requires no modifications. However, the file contains comments that explain how to change it to control how users are linked, including linking users in multiple SULs.

```
<?xml version="1.0" encoding="UTF-8"?>

<!--=================================================================

     Copyright &copy 2003 Sun Microsystems, Inc. All rights reserved.

     Patents Pending.

     SUN PROPRIETARY/CONFIDENTIAL.

     Use is subject to license terms.

===================================================================-->


<!-- This xml file is used to drive an ILODE (Initial Linking Of Directory
     Entries) operation from the command line.  It is passed to the
     LinkUsers script as the -f option.
-->

<UserLinkingOperationList>

  <!-- UserLinkingOperation encapsulates the configuration of a
       single SUL to ILODE. It includes the SUL ID and a list of
       attributes to match.  A separate UserLinkingOperation must
       be specified for each SUL being ILODE’d. -->
  <UserLinkingOperation parent.attr="UserLinkingOperation" sulid="SUL1">

    <!-- UserMatchingCriteria encapsulates a list of attributes
         that must match for a user to be linked. -->


    <!-- For two users to match using this UserMatchingCriteria, they
         must have the same givenName and the same sn. -->
    <UserMatchingCriteria parent.attr="UserMatchingCriteria">
      <AttributeMap parent.attr="AttributeMap">
        <AttributeDescription parent.attr="SunAttribute" name="sn"/>
        <AttributeDescription parent.attr="WindowsAttribute" name="sn"/>
      </AttributeMap>
        <AttributeMap parent.attr="AttributeMap">
          <AttributeDescription parent.attr="SunAttribute" name="givenName"/>
          <AttributeDescription parent.attr="WindowsAttribute" name="givenName"/>
        </AttributeMap>
      </UserMatchingCriteria>

    <!-- Multiple UserMatchingCriteria can be specified for a single
         SUL.  They are treated as a logical OR.  In this example, (the
         givenName’s and sn’s must match (see above)) OR (the employee(Number|ID)
must
         match), for the user to be linked.  Notice that attribute that
         is specified, employeeNumber, is the name of the DS
         attribute. -->

    <!-- This UserMatchingCriteria is commented out because employeeNumber is not
        an indexed attribute in DS.  All attributes used in a UserMatchingCriteria
         should be indexed.
    <UserMatchingCriteria parent.attr="UserMatchingCriteria">
      <AttributeMap parent.attr="AttributeMap">
        <AttributeDescription parent.attr="SunAttribute" name="employeeNumber"/>
        <AttributeDescription parent.attr="WindowsAttribute" name="employeeID"/>
      </AttributeMap>
     </UserMatchingCriteria>

  </UserLinkingOperation>

  <!-- When multiple SULs are ILODE’d, a separate UserLinkingOperation
       is specified for each.  As shown here, each UserLinkingOperation
       can use different UserMatchingCriteria: in this example, users in
       SUL2 are only linked if their sn and employeeNumber match.
       Note: this UserLinkingOperation is currently commented out because
       the example configuration only has a single SUL.
  <UserLinkingOperation parent.attr="UserLinkingOperation" sulid="SUL2">
    <UserMatchingCriteria parent.attr="UserMatchingCriteria">
      <AttributeMap parent.attr="AttributeMap">
        <AttributeDescription parent.attr="SunAttribute" name="sn"/>
        <AttributeDescription parent.attr="WindowsAttribute" name="sn"/>
      </AttributeMap>
        <AttributeMap parent.attr="AttributeMap">
         <AttributeDescription parent.attr="SunAttribute" name="employeeNumber"/>
         <AttributeDescription parent.attr="WindowsAttribute" name="employeeID"/>
      </AttributeMap>
    </UserMatchingCriteria>
  </UserLinkingOperation>
  -->

</UserLinkingOperationList>
```

Appendix C

Running Services as Non-Root

Identity Synchronization for Windows 1.0 requires root privileges to install and run its services. If you wish to run services under a non-root user perform the following:

1. Optionally use the UNIX `useradd` command to create a user account for Identity Synchronization for Windows.
   (You also can use nobody user to run services.)
2. If you are going to install a Sun ONE Directory Server connector on Solaris, you must choose a non-privileged port for the connector during installation. For example, ports larger than 1024 are acceptable.
3. After installing all components, use the `/etc/init.d/isw stop` command as root to shut down Identity Synchronization for Windows.
4. You must update the ownership of some files. Consider an example where the product was installed under `/usr/sunone/servers/isw-example`.
   a. As root, execute the following commands:

   ```
   cd /usr/sunone/servers/isw-example
   chown -R idsync logs/CNN* resources etc persist
   chown idsync logs
   ```

   b. If the core is installed on this host, execute the following command as root:

   ```
   chown idsync resources/SystemManagerBootParams.cfg
   resources/CentralLoggerManagerInitParams.cfg
   ```

   c. If there are any connectors are installed on this host, execute the following command as root:

   ```
   chown -R idsync resources/connectors
   ```

   d. If the plugin is installed on Solaris and the Sun ONE Directory Server is running with non-root privileges, then make sure that the log directory of the plugin is writable by the user account the directory server is running as. For example, if the Directory Server is running as user `sunds`, the following command-line should be executed:

```
chown -R sunds /usr/sunone/servers/isw-example/logs/SUBC*
```

5. By default, the start-up and shut-down scripts expect the `pid` file to reside in the installation root. To avoid having to make the installation root directory writable by the Identity Synchronization for Windows user, you must move the `pid` file to a directory that is more suitable, such as the `logs` directory.

   a. As root, execute the following commands:

   ```
   cd /usr/sunone/servers/isw-example
   ```

   ```
   perl -p -i -e ’s/pid.txt/logs\pid.txt/g’ *_watchdog.sh
   /etc/init.d/isw
   ```

   b. Open the `/etc/init.d/isw` file in a text editor and replace the `"$EXEC_START_WATCHDOG" "$JAVA_PATH" "$PSW_HOME"` line with the following line:

   ```
   su idsync -c "$EXEC_START_WATCHDOG '$JAVA_PATH' '$PSW_HOME'"
   ```

6. As root, use the following command to restart the service:

   ```
   /etc/init.d/isw start
   ```

7. Use the following command to verify that the components are running under the userid of the assigned user:

   ```
   ps -ef | grep idsync
   ```

Appendix D

Synchronization User List Definitions and Configuration

This appendix contains information on synchronized user list definitions and multiple domain configuration information in the following sections:

- Understanding Synchronized User List Definitions
- Configuring Multiple Windows Domains

Understanding Synchronized User List Definitions

Every Synchronized User List contains two definitions that identify which users in a directory to synchronize, which users to exclude from synchronization, and where to create new users. One definition identifies which Sun ONE Directory Server users to synchronize and the other identifies the Windows users to synchronize.

The following table describes the components of an SUL definition:

Table 10-4 SUL Definition Components

Component	Definition	Applicable		
		Sun	AD	NT
Base DN	Defines the parent LDAP node of all users to be synchronized. A synchronization scope's base DN includes all users under that DN — unless the users are excluded by the synchronization scope's filter or the user's DN is matched in a more specific synchronization scope. For example, `ou=sales,dc=example,dc=com`.	Yes	Yes	No

Table 10-4 SUL Definition Components

Component	Definition	Applicable		
		Sun	AD	NT
Filter	Defines an LDAP-like filter that is used to include or exclude users from a synchronization scope. For example, `(& (employeeType=manager)(st=CA))` will include managers in California only.	Yes	Yes	Yes
Creation Expression	Defines the parent DN and naming attribute of newly created users. The creation expression must include the base DN of the synchronization scope. For example, `cn=%cn%,ou=sales,dc=example,dc=com`. (Where the `%cn%` token is replaced with a value from the user entry being created.)	Yes	Yes	No

NOTE To synchronize users in a Sun ONE Directory Server with multiple Active Directory domains, you must define one SUL for each Active Directory domain.

If multiple SULs are defined, the program determines membership in an SUL by iteratively matching each SUL definition. SUL definitions with more specific base DNs will be examined first. For example, a match against `ou=sales,dc=example,dc=com` will be tested before `dc=example,dc=com`.

If two SUL definitions have the same base DN and different filters, then the program cannot determine automatically which filter should be tested first, so the administrator must use the Resolve Domain Overlap to order the two SUL definitions. If a user matches the base DN of an SUL definition but does not match any filters for that base DN, then the user will be excluded from synchronization — even if that user matches the filter for a less specific base DN.

Configuring Multiple Windows Domains

To support synchronizing multiple Windows domains to the same Sun ONE Directory Server container (e.g. `ou=people,dc=example,dc=com`) Identity Synchronization for Windows has introduced synthetic Windows attributes that contain domain information.

- For Active Directory domains, Identity Synchronization for Windows sets the `activedirectorydomainname` attribute to the Active Directory domain name (e.g. east.example.com) before synchronizing the entry to the Sun ONE Directory Server.
- For Windows NT domains, Identity Synchronization for Windows sets the `user_nt_domain_name` attribute to the Windows NT domain name (e.g. NTEXAMPLE) before synchronizing the entry to the Sun ONE Directory Server.

While these attributes do not actually appear on the Windows user entries, they are available for synchronization in the Identity Synchronization for Windows console and can be mapped to a Sun ONE Directory Server user attribute. Once the domain attributes are mapped, they will be set in the Sun ONE Directory Server entries during synchronization and can be used in Synchronization User List (SUL) filters.

The following example illustrates how these attributes are used. This example assumes that three Windows domains (two Active Directory domains and one Windows NT domain) will be synchronized with a single Sun ONE Directory Server instance.

1. Users in the Active Directory domain east.example.com will be synchronized to the Sun ONE Directory Server under `ou=people,dc=example,dc=com`.
2. Users in the Active Directory domain west.example.com will be synchronized to the Sun ONE Directory Server under `ou=people,dc=example,dc=com`.
3. Users in the Windows NT domain NTEXAMPLE will be synchronized to the Sun ONE Directory Server under `ou=people,dc=example,dc=com`.

When you create or modify a Sun ONE Directory Server user, Identity Synchronization for Windows uses the SUL filters to determine to which Windows domain the user should be synchronized (because each Sun ONE Directory Server SUL has the same base DN, `ou=people,dc=example,dc=com`). The `activedirectorydomainname` and `user_nt_domain_name` attributes make constructing these filters easy.

In the attributes panel on the Identity Synchronization for Windows console, first map the Sun ONE Directory Server destinationindicator attribute to the Active Directory activedirectorydomainname attribute and to the Windows NT user_nt_domain_name attribute.

Then, configure one SUL for each Windows domain as follows. Notice that each Sun ONE Directory Server SUL definition has the same base DN and creation expression, but the filters indicate the domain of the corresponding Windows user entry.

```
EASTSUL

  Sun ONE Directory Server definition
    Base DN:  ou=people,dc=example,dc=com
    Filter:  destinationindicator=east.example.com
    Creation Expression:  cn=%cn%,ou=people,dc=example,dc=com
Active Directory definition (east.example.com)
    Base DN:  cn=users,dc=east,dc=example,dc=com
    Filter:  <none>
    Creation Expression:  cn=%cn%,cn=users,dc=east,dc=example,dc=com
WEST_SUL
  Sun ONE Directory Server definition
    Base DN:  ou=people,dc=example,dc=com
    Filter:  destinationindicator=west.example.com
    Creation Expression:  cn=%cn%,ou=people,dc=example,dc=com
Active Directory definition (west.example.com)
    Base DN:  cn=users,dc=west,dc=example,dc=com
    Filter:  <none>
    Creation Expression:  cn=%cn%,cn=users,dc=west,dc=example,dc=com
NT_SUL
  Sun ONE Directory Server definition
    Base DN:  ou=people,dc=example,dc=com
    Filter:  destinationindicator=NTEXAMPLE
    Creation Expression:  cn=%cn%,ou=people,dc=example,dc=com
Active Directory definition (NTEXAMPLE)
    Base DN:  NA
    Filter:  <none>
    Creation Expression:  NA
```

To understand how these settings allow Sun ONE Directory Server user entries to synchronize with separate Windows domains, consider this test case:

1. Create cn=Jane Test,cn=users,dc=east,dc=example,dc=com in the Active Directory east.example.com domain.
2. Identity Synchronization for Windows creates the user entry cn=Jane Test,ou=people,dc=example,dc=com in the Sun ONE Directory Server with destinationindicator=east.example.com.

3. Modify the `cn=Jane Test,ou=people,dc=example,dc=com` entry in the Sun ONE Directory Server.

4. Because Jane Test's `destinationindicator` attribute is east.example.com, her entry will match the EAST_SUL Synchronization User List filter, and the modification will be synchronized to the east.example.com Active Directory domain.

This example assumes user creations are synchronized from Windows to the Sun ONE Directory Server. If this is not the case, you also can run the idsync resync command to set the `destinationindicator` attribute. The example uses an existing attribute on `inetorgperson`, `destinationIndicator`, which might be used for other purposes. If this attribute is already in use or a different `objectclass` is selected, you must map some attribute in the user's Sun ONE Directory Server entry to the `user_nt_domain_name` and/or the `activedirectorydomainname` attribute(s). The Sun ONE Directory Server attribute you choose to hold this value must be in the `objectclass` you are using for the rest of the attribute mapping configuration. If there is no unused attribute to hold this domain information, you must create a new `objectclass` to include a new domain attribute and all other attributes you will be using with Identity Synchronization for Windows.

Appendix E

Installation Notes for Replicated Environments

This Appendix gives brief overviews of the steps required to configure and secure a Multi-master replication (MMR) deployment in the following sections:

- Summary of Steps for Configuring Replication
- Replication Over SSL

NOTE This is only an overview for your convenience. Designing and implementing an MMR deployment is *complex*. Refer to the *Sun ONE Directory Server 5.2 Deployment Guide* to plan your deployment and the *Sun ONE Directory Server 5.2 Administrator's Guide* to implement it.

Summary of Steps for Configuring Replication

Identity Synchronization for Windows 1.0 supports replicating a single suffix. To configure any replication topology, you should proceed in the following order:

1. Define your replication manager entry on all servers except single masters. Or simply decide to use the default replication manager on all servers.
2. On all servers containing a dedicated consumer replica:
   a. Create an empty suffix for the consumer replica.
   b. Enable the consumer replica on the suffix through the replication wizard.
   c. Optionally, configure the advanced replica settings.
3. On all servers containing a hub replica, if applicable:

   a. Enable the hub replica on the suffix through the replication wizard.
   b. Optionally, configure the advanced replica settings.

4. On all servers containing a master replica:
   a. Choose a suffix on one of the masters that will be the master replica.
   b. Enable the master replica on the suffix through the replication wizard.
   c. Optionally, configure the advanced replica settings.

5. Configure the replication agreements on all supplier replicas, in the following order:
   a. Between masters in a multi-master set.
   b. Between masters and their dedicated consumers.
   c. Between masters and hub replicas.

   Optionally, you may configure fractional replication and initialize the consumer and hub replicas at this stage. In the case of multi-master replication, initialize all masters from the same master replica containing the original copy of the data.

6. Configure replication agreements on all hub replicas supplied directly from a master. These agreements are between the hub replicas and their consumers. Optionally, you may initialize the consumer replicas at this stage. Repeat this step for every level of hubs in your cascading replication.

NOTE It is very important to create and configure all replicas before you attempt to create a replication agreement. This also allows you to initialize consumer replicas immediately after you create the replication agreement. Consumer initialization is always the last stage in setting up replication.

Replication Over SSL

You can configure Directory Servers involved in replication so that all replication operations occur over an SSL connection. To do so, complete the following steps:

NOTE All references in the following procedure refer to chapters in the *Sun ONE Directory Server 5.2 Administrator's Guide*

1. Configure both the supplier and consumer servers to use SSL.

   Refer to Chapter 11 "Implementing Security" for details.

NOTE Replication over SSL will fail if the supplier server certificate is:

- A self-signed certificate.
- An SSL server-only certificate that cannot act as a client during an SSL handshake.

2. If replication is not configured for the suffix on the consumer server, enable it as described in "Enabling a Consumer Replica".
3. Follow the procedure in "Advanced Consumer Configuration", to define the DN of the certificate entry on the consumer as another replication manager.
4. If replication is not configured for the suffix on the supplier server, enable it as described in "Enabling a Hub Replica", or "Enabling a Master Replica".
5. On the supplier server, create a new replication agreement to send updates to the consumer on the secure SSL port. Follow the procedure in "Creating Replication Agreements", for detailed instructions. Specify a secure port on the consumer server and select the SSL option of either using a password or a certificate. Enter a DN for the SSL option that you chose, either a replication manager or a certificate.

After you finish configuring the replication agreement, the supplier will send all replication update messages to the consumer over SSL and will use certificates if you chose that option. Customer initialization will also use a secure connection if performed through the console using an agreement configure for SSL.

Glossary

attribute Holds descriptive information about an entry. Attributes have a label and a value. Each attribute also follows a standard syntax for the type of information that can be stored as the attribute value.

attribute list A list of required and optional attributes for a given entry type or object class.

Audit Log A log file that contains entries for day-to-day events such as a user's password being synchronized. The administrator can control the number and detail of entries in this log by changing the log level via the Identity Synchronization for Windows Console. Each Connector has an audit log for users processed by that Connector. In addition, there is a centralized audit log that is the aggregation of every Connectors' audit logs.

authenticating directory server In pass-through authentication (PTA), the authenticating directory server is the directory server that contains the authentication credentials of the requesting client. The PTA-enabled host sends PTA requests it receives from clients to the bind host.

authentication Process of proving the identity of the client user to the Directory Server. Users must provide a bind DN and the corresponding password in order to be granted access to the directory. Directory Server allows the user to perform functions or access files and directories based on the permissions granted to that user by the directory administrator.

authentication certificate Digital file that is not transferable, not forgeable, and is issued by a third party. Authentication certificates are sent from server to client or client to server in order to verify and authenticate the other party.

base DN Base distinguished name. A search operation is performed on the base DN, the DN of the entry, and all entries below it in the directory tree.

base distinguished name *See base DN.*

bind DN Distinguished name used to authenticate to Directory Server when performing an operation.

bind distinguished name *See bind DN.*

bind rule In the context of access control, the bind rule specifies the credentials and conditions that a particular user or client must satisfy in order to get access to directory information.

branch entry An entry that represents the top of a subtree in the directory.

browser Software, such as Netscape Navigator, used to request and view World Wide Web material stored as HTML files. The browser uses the HTTP protocol to communicate with the host server.

CA *See Certificate Authority.*

cascading replication In a cascading replication scenario, one server, often called the hub supplier, acts both as a consumer and a supplier for a particular replica. It holds a read-only replica and maintains a change log. It receives updates from the supplier server that holds the master copy of the data, and in turn supplies those updates to the consumer.

Central Logs An aggregation of every Connectors' audit and error logs. An administrator can monitor the health of an entire Identity Synchronization for Windows installation by only monitoring these logs. These logs can be viewed directly or via the Console.

certificate A collection of data that associates the public keys of a network user with their DN in the directory. The certificate is stored within the directory as user object attributes.

Certificate Authority Company or organization that sells and issues authentication certificates. You may purchase an authentication certificate from a Certificate Authority that you trust. Also known as a CA.

character type Distinguishes alphabetic characters from numeric or other characters and the mapping of upper-case to lower-case letters.

Connector subcomponent (subcomponent): a lightweight process that runs separate from a

Connector A subcomponent runs close to the directory source that a Connector manages, and enables functionality in the Connector that cannot be achieved in a remote machine or separate process. The subcomponent communicates with Connector over an SSL connection. (See Windows NT Change Detector and Sun ONE Directory Server Plugin).

client *See LDAP client.*

daemon A background process on a UNIX machine that is responsible for a particular system task. Daemon processes do not need human intervention to continue functioning.

Directory Access Protocol *See DAP.*

Directory Source A Sun ONE Directory Server, a Windows 2000 Active Directory, or a Windows NT SAM Registry.

directory tree The logical representation of the information stored in the directory. It mirrors the tree model used by most file systems, with the tree's root appearing at the top of the hierarchy. Also known as DIT.

Directory Manager The privileged directory server administrator, comparable to the root user in UNIX.

directory service A database application designed to manage descriptive, attribute-based information about people and resources within an organization.

distinguished name String representation of an entry's name and location in an LDAP directory.

DIT *See directory tree.*

DNS Domain Name System. The system used by machines on a network to associate standard IP addresses (such as 198.93.93.10) with hostnames (such as `www.iPlanet.com`). Machines normally get the IP address for a hostname from a DNS server, or they look it up in tables maintained on their systems.

DNS alias A DNS alias is a hostname that the DNS server knows points to a different host—specifically a DNS CNAME record. Machines always have one real name, but they can have one or more aliases. For example, an alias such as `www.[yourdomain].[domain]` might point to a real machine called `realthing.[yourdomain].[domain]` where the server currently exists.

Error Log A log file that contains error and warning messages that demand attention. An error is a severe condition that prevents Identity Synchronization for Windows from operating correctly. A warning is a less sever condition, such as a single password update failing. Messages in the error log also appear in the audit log to facilitate diagnosing the problem. Each Connector has an error log for users processed by that Connector. In addition, there is a centralized error log that is the aggregation of every Connectors' error logs.

file extension The section of a filename after the period or dot (.) that typically defines the type of file (for example, .GIF and .HTML). In the filename `index.html` the file extension is `html`.

file type The format of a given file. For example, graphics files are often saved in GIF format, while a text file is usually saved as ASCII text format. File types are usually identified by the file extension (for example, .GIF or .HTML).

Java Message Service (JMS) A standardized API for handling asynchronous messaging between Java applications. It's publish/subscribe model separates producers of information from its consumers via topics.

hostname A name for a machine in the form machine.domain.com, which is translated into an IP address. For example, `www.example.com` is the machine www in the subdomain example and domain com.

HTML Hypertext Markup Language. The formatting language used for documents on the World Wide Web. HTML files are plain text files with formatting codes that tell browsers such as the Netscape Navigator how to display text, position graphics and form items, and display links to other pages.

HTTP Hypertext Transfer Protocol. The method for exchanging information between HTTP servers and clients.

Identity Synchronization for Windows **Components** The pieces that compose a complete Identity Synchronization for Windows installation. This includes the Connectors, Connector subcomponents, the Console, the Registry, the Sun ONE MQ Broker, and the central logs.

Identity Synchronization for Windows **Connector** A process that manages synchronizing users in a single data source type. A Connector can manage multiple data sources of the same type (e.g. Active Directories). It is responsible for detecting user changes in the data source, publishing these changes to remote

Connectors over the Sun ONE Message Queue, subscribing to user change topics, and applying updates from these topics to the data source. (See also Windows Active Directory Connector, Sun ONE Directory Server Connector, Windows NT Connector).

Identity Synchronization for Windows **Console** Identity Synchronization for Windows's user interface that allows an Administrator to configure Identity Synchronization for Windows settings and monitor the status of an entire Identity Synchronization for Windows deployment.

Identity Synchronization for Windows **Core** The Identity Synchronization for Windows components other than the Connectors and the Connector subcomponents. This includes the Console, the Registry, the Sun ONE MQ Broker, and the central logs.

Identity Synchronization for Windows **Registry** An LDAP directory that stores the complete Identity Synchronization for Windows configuration and the status of every component.

International Standards Organization *See ISO.*

IP address Internet Protocol address. A set of numbers, separated by dots, that specifies the actual location of a machine on the Internet (for example, 192.168.2.1).

ISO International Standards Organization.

LDAP Lightweight Directory Access Protocol. Directory service protocol designed to run over TCP/IP and across multiple platforms.

LDAP client Software used to request and view LDAP entries from an LDAP Directory Server.

LDAP URL Provides the means of locating directory servers using DNS and then completing the query via LDAP. A sample LDAP URL is `ldap://ldap.iplanet.com`

LDIF LDAP Data Interchange Format. Format used to represent Directory Server entries in text form.

LDIF entry A group of lines in the LDIF file that contains information about an object.

Lightweight Directory Access Protocol *See LDAP.*

locale Identifies the collation order, character type, monetary format, and time / date format used to present data for users of a specific region, culture, and/or custom. This includes information on how data of a given language is interpreted, stored, or collated. The locale also indicates which code page should be used to represent a given language.

NIS Network Information Service. A system of programs and data files that UNIX machines use to collect, collate, and share specific information about machines, users, file systems, and network parameters throughout a network of computers.

On Demand Password Update A mechanism whereby a user's password in Sun ONE Directory Server is not updated until the user attempts to authenticate to the Sun ONE Directory. The user's password is synchronized only if the provided password matches what is stored in the Windows environment. This simplifies the AD and NT Connectors because they are not required to capture clear text passwords.

password file A file on UNIX machines that stores UNIX user login names, passwords, and user ID numbers. It is also known as `/etc/passwd`, because of where its located.

password policy A set of rules that govern how passwords are used in a given directory.

permission In the context of access control, the permission states whether access to the directory information is granted or denied, and the level of access that is granted or denied. See access rights.

protocol A set of rules that describes how devices on a network exchange information.

proxy authentication A special form of authentication where the user requesting access to the directory does not bind with its own DN but with a proxy DN.

proxy DN Used with proxied authorization. The proxy DN is the DN of an entry that has access permissions to the target on which the client-application is attempting to perform an operation.

root The most privileged user available on UNIX machines (also called superuser). The root user has complete access privileges to all files on the machine.

root suffix The parent of one or more subsuffixes. A directory tree can contain more than one root suffix.

schema Definitions describing what types of information can be stored as entries in the directory. When information that does not match the schema is stored in the directory, clients attempting to access the directory may be unable to display the proper results.

schema checking Ensures that entries added or modified in the directory conform to the defined schema. Schema checking is on by default and users will receive an error if they try to save an entry that does not conform to the schema.

Secure Sockets Layer *See SSL.*

Server Console Java-based application that allows you to perform administrative management of your Directory Server from a GUI.

server root A directory on the server machine dedicated to holding the server program and configuration, maintenance, and information files.

service A background process on a Windows NT machine that is responsible for a particular system task. Service processes do not need human intervention to continue functioning.

Simple Network Management Protocol *See SNMP.*

SNMP Simple Network Management Protocol. Used to monitor and manage application processes running on the servers, by exchanging data about network activity.

SSL Secure Sockets Layer. A software library establishing a secure connection between two parties (client and server) used to implement HTTPS, the secure version of HTTP.

suffix The name of the entry at the top of the directory tree, below which data is stored. Multiple suffixes are possible within the same directory. Each database has only one suffix.

Sun ONE Directory Server Connector The Connector that manages synchronizing users in Sun ONE Directory Servers. A single Sun ONE Directory Server Connector can synchronize multiple Sun ONE Directory Servers with Active Directories or NT SAM Registries.

Sun ONE Directory Server Plugin A subcomponent of the Sun ONE Directory Server Connector that runs in the Sun ONE Directory Server. The Sun ONE Directory Server Plugin enables On Demand Password Updates and encrypts password change events in the retro change log so they can be retrieved by the Sun ONE Directory Server Connector.

Sun ONE Message Queue (Sun ONE MQ): Sun ONE Message Queue is an implementation of the Java Message Service (JMS) specification. (See also Java Message Service, Sun ONE Message Queue Broker, Sun ONE Message Queue Message, Sun ONE Message Queue Publisher, Sun ONE Message Queue Subscriber, and Sun ONE Message Queue Topic).

Sun ONE Message Queue Broker A server that links Sun ONE Message Queue Publishers to Sun ONE Message Queue Subscribers. The broker persistently stores messages that a publisher sends, and delivers them to all subscribers, even if they are not available when the message is published. The broker authenticates all publishers and subscribers before permitting access to topics and messages.

Sun ONE Message Queue Message A packet of information that a publisher sends to a topic and that subscribers receive from a topic. In Identity Synchronization for Windows, all user updates are sent as messages.

Sun ONE Message Queue Publisher (publisher) A producer of messages sent to a specific topic. In Identity Synchronization for Windows, a Connector publishes messages to a topic based on which Synchronization User List the user is located in.

Sun ONE Message Queue Subscriber (subscriber): A consumer of messages that were sent to a specific topic. In Identity Synchronization for Windows, a Connector subscribes to messages based on the Synchronization User Lists that it manages.

Sun ONE Message Queue Topic (topic) The link between publishers of information and the subscribers interested in that information. Sun ONE Identity Synchronization for Windows (Identity Synchronization for Windows): a product that securely synchronizes password values bi-directionally between Sun ONE Directory Server and Windows directories, namely Windows 2000 Active Directories and NT SAM Registries.

Synchronization Scope Definition Rules that define user membership in a Synchronization User List for a single data source. The Synchronization Scope Definition includes a location (e.g. a base-dn or ou) and a filter (e.g. "country=US and (not member-of Administrators)"). Each Synchronization Scope Definition is managed by a single Connector. However, a Connector may manage multiple Synchronization Scope Definitions.

Synchronization User List A grouping of users that are present in both a Sun ONE Directory Server and Windows data source whose passwords are being synchronized. Each user is in a single Synchronization User List. Identity Synchronization for Windows uses a user's Synchronization User List membership to determine if a given user is being synchronized, and if they are, the location of the user entry in the remote directory. A Synchronization User List includes a Synchronization Scope Definition for the users in the Sun ONE Directory Server environment and a Synchronization Scope Definition for the users in the Windows environment (either Active Directory or NT SAM Registry).

superuser The most privileged user available on UNIX machines (also called root). The superuser has complete access privileges to all files on the machine.

target In the context of access control, the target identifies the directory information to which a particular ACI applies.

TCP/IP Transmission Control Protocol/Internet Protocol. The underlying network protocol for the Internet.

time / date format Indicates the customary formatting for times and dates in a specific region.

TLS Transport Layer Security. The new standard for secure socket layers, a public key based protocol.

tModel A common structure that provides information about a business. The information that makes up a tModel includes a key, a name, an optional description, and a URL that points to a location for additional information.

topology The way a directory tree is divided among physical servers and how these servers link with one another.

Transport Layer Security **uid** A unique number associated with each user on a UNIX system.

UNSPSC Universal Standard Products and Service Codes. A set of product and service classifications.

URL Uniform Resource Locator. The addressing system used by the server and the client to request documents. It is often called a location. The format of a URL is `[protocol]://[machine:port]/[document]`. The port number is necessary only on selected servers, and it is often assigned by the server, freeing the user of having to place it in the URL.

Windows Active Directory Connector (AD Connector) A Connector that manages synchronizing users in Windows 2000 Active Directories. A single AD Connector can synchronize multiple AD Domains with a Sun ONE Directory Server.

Windows NT Change Detector (NT Change Detector) A subcomponent of the Windows NT Connector that monitors the NT Security Event Log to determine when a user’s password has changed. This enables immediate synchronization of user entries. The NT Change Detector must run on the Primary Domain Controller.

Windows NT Connector (NT Connector) A Connector that manages synchronizing users in Windows NT SAM Registries. This Connector must run on a Windows NT machine. (See also Windows NT Change Detector).

Index