CA™ SOA Security Manager

Policy Server Installation Guide

r12.1

Second Edition

CA Product References

This document references the following CA products:

- CA™ SOA Security Manager
- CA™ SiteMinder® Web Access Manager

Contact CA

Contact Technical Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At http://ca.com/support, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short customer survey, which is also available on the CA Support website, found at http://ca.com/docs.

Contents

Chapter 8: Installing the Administrative UI 243

Chapter 1: Installation Overview

This section contains the following topics:

Intended Audience

This guide is intended for users who have a working knowledge of:

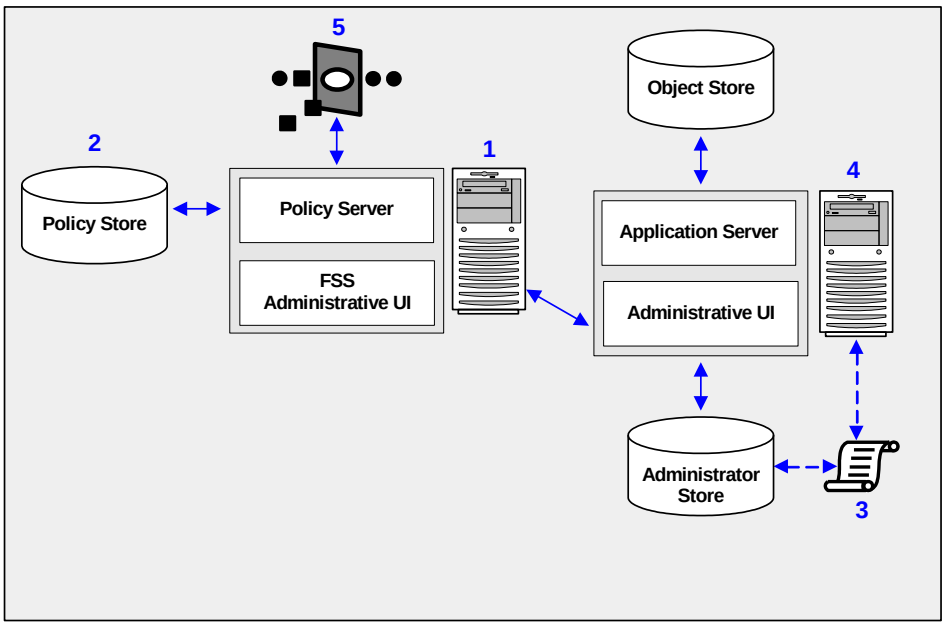
- directory servers
- relational databases
- Web servers

This guide assumes you are familiar with Java, J2EE standards, and application server technology, and that you have the following technical knowledge:

- An understanding of J2EE application servers and multi-tier architecture.
- Experience with managing an application server.

SOA Security Manager Installation Roadmap

Installing SOA Security Manager requires you to install and configure several components, all of which are shown in the following diagram of a basic sample installation:



You should install and configure the SOA Security Manager components in the order shown in the sample illustration, as follows:

1. **Policy Server and FSS Administrative UI**
   - The *Policy Server* provides policy management, authentication, authorization, and accounting services.
   - The *FSS Administrative UI* is installed with the Policy Server and is only used to manage eTrust SiteMinder FSS features. If you will not need to generate WS-Security SAML tokens, use of the FSS Administrative UI is not required. Although part of the core Policy Server installation, the FSS Administrative UI must be registered with the Policy Server before it may be used. Registering the FSS Administrative UI is completed through the Administrative UI. Therefore, you must install and configure the Administrative UI before registering the FSS Administrative UI.
2. **Policy store** - The *policy store* contains all of the Policy Server data. You can configure a policy store in a supported LDAP or relational database.

3. **Directory XML file**—A *directory xml file* (directory.xml) describes how objects such as users, groups, and organizations are stored in your administrator user store. Prior to installing the Administrative UI, you configure a supported directory server or database-specific directory XML file so that the Administrative UI is able to locate your administrative users in the administrator user store.

4. **Administrative UI**—You use the Administrative UI to manage SOA Security Manager administrator accounts, objects, and policy data through the Policy Server. You configure a directory XML file, an administrator user store, and an object store when installing the Administrative UI:

   - **Object store**—The Administrative UI is an asynchronous application that is event and task-based. The object store stores information about these tasks and events. You configure an object store in either a MS SQL Server or Oracle database.
   - **Administrator user store**—The Administrative UI authenticates SOA Security Manager administrator accounts using the administrator user store. All of your administrator accounts must be stored in a single administrator user store. You configure an administrator user store in a supported LDAP directory server or ODBC database when installing the Administrative UI.

     Note: If you are upgrading to r12.1, you may use an existing user store as an administrator store.

   Note: CA recommends installing the Administrative UI on a system that is not hosting the Policy Server.

5. **SOA Agent (or SOA Security Gateway)**—A *SOA Agent* is integrated with a web server or application server. The Agent lets SOA Security Manager manage access to web services according to predefined security policies. A SOA Agent is also integrated into the SOA Security Gateway, which provides XML gateway functionality in addtion to SOA Security Manager access control.

   Note: CA recommends installing SOA Agents and SOA Security Gateways on systems that are not hosting the Policy Server.

More information:

Installing the Policy Server on Windows Systems (see page 29)
Installing the Policy Server on UNIX Systems (see page 55)
Administrative User Interfaces Overview (see page 16)
Registering the FSS Administrative UI (see page 299)
LDAP Directory Servers as a Policy or Key Store (see page 89)
Relational Databases as a Policy or Key Store (see page 131)
Installing the Administrative UI (see page 243)

Note: More information on installing a SOA Agent exists in the *SOA Security Manager Implementation Guide*.

Administrative User Interfaces Overview

There are two graphical user interfaces (UIs), which configure specific SiteMinder policy objects, as follows:

- **SOA Security Manager Administrative UI** (Administrative UI)—The Administrative UI is a web-based administration console that is installed independent of the Policy Server. The Administrative UI is the tool for configuring the majority of tasks related to access control, such as authentication and authorization policies, Enterprise Policy Management (EPM), reporting and policy analysis.

  Use the Administrative UI to create, view, modify, and delete all Policy Server objects except those related to eTrust SiteMinder FSS. All federation-related configuration tasks should be handled using the FSS Administrative UI.

- **SiteMinder Federation Security Services Administrative UI** (FSS Administrative UI)—The FSS Administrative UI is an applet-based application that is installed with the Policy Server. Use the FSS Administrative UI only to configure affiliates required to generate WS-Security SAML assertion tokens and SOA Security Manager eTrust SiteMinder FSS.

  The intent of the FSS Administrative UI is to let you manage SOA Security Manager eTrust SiteMinder FSS. If you are familiar with previous versions of the SiteMinder Policy Server User Interface, you will notice that all SOA Security Manager objects appear in the FSS Administrative UI, except the application objects for Enterprise Policy Management (EPM). You should *not* use the FSS Administrative UI to manage these objects. If you need information while using the FSS Administrative UI, please consult the online help.

  Note: If your organization is not federating with a partner, you may safely leave the eTrust SiteMinder FSS on the Policy Server machine without registering it.

Pre-Installation Checklist

You may want to print the following to use as a checklist to help ensure you meet all of the necessary system and software requirements before installing a Policy Store and the Administrative UI.

- ☐ Install the SOA Security Manager documentation.
- ☐ Confirm that the Windows or UNIX system that is to host the Policy Server meets the minimum system requirements. Refer to Policy Server System Requirements.
- ☐ Confirm that the Windows or UNIX system that is to host the Administrative UI meets the minimum system requirements. Refer to Administrative User Interface System Requirements.
- ☐ Confirm that a supported application server is installed on the system that is to host the Administrative UI. Refer to Application Server Requirements.
- ☐ Confirm that your environment meets the required directory server and database requirements for the Administrative UI and reporting feature. Refer to Directory Server and Database Requirements.

Scripting Interface

The Command Line Interface allows you to write Perl scripts to configure and manage policy stores. The installation program installs a full version of Perl and puts the interface files in the *siteminder_installation*/CLI directory.

siteminder_installation

Specifies the installed location of SOA Security Manager.

Example: /home/smuser/siteminder/CLI

To use the Command Line Interface, make sure the following directory is in your system's PATH environment variable before any other Perl bin directories on your machine.

For example: /home/smuser/siteminder/CLI/bin

Note: More information on the scripting interface exists in the *Programming Guide for Perl*

Chapter 2: Policy Server Installation Requirements

This section contains the following topics:

Windows Requirements

The Windows system to which you are installing the Policy Server must meet at least the following system requirements:

- **CPU**—Intel Pentium III or better.
- **Memory**—512 MB system RAM.
- **Available disk space**:
  - 270 MB free disk space in the install location.
  - 180 MB of free space in the system's temporary file location.

    Note: These requirements are based on a medium size policy database of approximately 1,000 policies.
- **JRE**—The required JRE version is installed on the system to which you are installing the Policy Server.
- **LDAP directory server or relational database—**ensure that LDAP directory server or relational database you plan on using as a policy store is supported.
- **Web server**—A supported Web server.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

UNIX Requirements

The UNIX system to which you are installing the Policy Server must meet at least the following system requirements:

- **Memory**—512 MB RAM.
- **Available disk space**:
  - 300 MB free disk space.
  - 200 MB free disk space in /tmp.

    Note: Typically, 10 MB or less free disk space in /tmp is required for the daily operation of the Policy Server. The Policy Server creates files and named pipes under /tmp. The path to which these files and pipes are created cannot be changed.
- **JRE**—The required JRE version is installed on the system to which you are installing the Policy Server.
- **LDAP directory server or relational database—**ensure that LDAP directory server or relational database you plan on using as a policy store is supported.
- **Web server**—A supported Web server.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Chapter 3: Administrative UI Installation Requirements

This section contains the following topics:

Administrative UI System Requirements

The following sections detail the minimum system requirements for installing the Administrative UI on a Windows and UNIX environment.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Windows

The Windows system to which you are installing the Administrative UI must meet the following minimum system requirements.

- **CPU**—Single or dual-processor, Intel Pentium III (or compatible), 700-900 MHz.
- **Memory**—512 MB system RAM. We recommend 1 GB.

  Note: If you are running WebSphere, 2 GB system RAM is required.

- **Available disk space**—540 MB.

  Note: If you are running WebSphere, 2 GB of available disk space is required.

- **Temp directory space**—450 MB.
- **JDK**—The required JDK version is installed on the system to which you are installing the Administrative UI.
- **Screen resolution**—1024 x 768 or higher resolution with 256 colors or better to view the Administrative UI properly.

UNIX

The UNIX system to which you are installing the Administrative UI must meet the following minimum system requirements.

- **CPU**
  - Solaris—Sparc Workstation 440 MHz.
  - Red Hat Linux—Single or dual-processor, Intel Pentium III (or compatible), 700-900 MHz.
- **Memory**—512 MB system RAM. We recommend 1 GB.

  Note: If you are running WebSphere, 2 GB system RAM is required.
- **Available disk space**—540 MB.

  Note: If you are running WebSphere, 2 GB of available disk space is required.
- **Temp directory space**—450 MB.
- **JDK**—The required JDK version is installed on the system to which you are installing the Administrative UI.
- **Screen resolution**—1024 x 768 or higher resolution with 256 colors or better to view the Administrative UI properly.

Application Server Requirements

The Administrative UI is a J2EE application and requires a SOA Security Manager supported application server. Ensure that:

- A supported version of JBoss, WebLogic, or WebSphere is installed on the system to which the Administrative UI is to be installed.
- The Administrative UI is the only application to be deployed on the application server.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

JBoss as an Application Server

A supported application server is required on the Administrative UI host system. Follow the installation instructions as described on the JBoss web site to install JBoss.

Important! Be sure to install JBoss in a directory pathname that contains no spaces.

WebLogic as an Application Server

The following sections provide basic instructions for using WebLogic as a SOA Security Manager application server.

Install WebLogic

Install a version of a WebLogic server that is supported by SOA Security Manager.

Note: More information on installing a WebLogic server exists in BEA's WebLogic server documentation.

Create a WebLogic Application Server Instance

Before installing the Administrative UI, create a WebLogic domain using the Configuration Wizard that is part of the WebLogic installation and do the following:

- Note the name of the domain. You will need the domain name when installing the Administrative UI.
- Select the Basic WebLogic Server Domain template.
- Verify that the JAVA_HOME variable is set to the path for the required Java environment in the setDomainEnv.cmd/ .sh file. This file is located in *web_logic_home*\user_projects\domains*weblogic_domain*\bin.

  web_logic_home

  Specifies the WebLogic server installation path.

  weblogic_domain

  Specifies the name of the WebLogic domain you created.

Verify the WebLogic Domain

Confirm the following:

- The WebLogic server is running.
- You can access the WebLogic console at http://*server.domain.port*/console

  Example: http://myserver.mycompany.com:7001/console
- In the WebLogic console, under Domain Configurations, select the domains link and verify that the WebLogic domain you created appears in the list of existing domains.

Note: Once you have completed the verification, shut down the application server to prepare for the Administrative UI installation.

WebSphere as an Application Server

The following sections provide basic instructions for using WebSphere as a SOA Security Manager application server.

New Topic (74)

Follow the WebSphere installation instructions as described in IBM's documentation.

When installing WebSphere, note the following:

- If you are installing the Administrative UI on Solaris, run the installation as root.

- If you are using WebSphere with Microsoft SQL Server 2005, enable XA transactions on Microsoft SQL Server 2005. The Administrative UI needs an XA data source for the database transactions to work properly.

  Note: For more information on enabling XA transactions on Microsoft SQL Server, go to http://msdn.microsoft.com/en-us/library/aa342335.aspx http://msdn.microsoft.com/en-us/library/aa342335.aspx.

- When using WebSphere on Windows, ensure that your Admin username is less than 12 characters long. If you have a username that is 12 characters or greater, the Administrative UI will not work. For example, the username "Administrator" is greater than 12 characters and will cause the Administrative UI to fail.
- Be sure to install WebSphere in a directory pathname that contains no spaces.

Verify WebSphere is Working

Use the snoop utility provided by IBM to verify that WebSphere is installed correctly before installing the Administrative UI.

To verify WebSphere is working

1. Enter http://<*fqdn:port*>/snoop to verify that WebSphere is installed correctly.

   Example: http:MyServer.MyCompany.com:9080/snoop.

   If WebSphere is installed correctly, Snoop Servlet—Request Client Information page is displayed in the browser.

2. Enter http://<*fqdn*>/snoop to verify that the WebSphere application server plug-in is installed correctly.

   Example: http://MyServer.MyCompany.com/snoop

   If WebSphere is installed correctly, the same Snoop Servlet—Request Client Information page is displayed in the browser.

   You have verified that WebSphere is working properly.

Note: Contact IBM Technical Support for additional assistance with WebSphere.

Set the Library Path for Solaris

This topic only applies if WebSphere is installed on a Solaris system. If your WebSphere application server is installed on a Solaris system, you must configure the WebSphere library path environment variable to successfully install the Administrative UI.

To set the WebSphere library path

1. Set the library path environment variable as follows:

   LD_LIBRARY_PATH=*IBM_home*/profiles/*profile_name*/installedApps/*cell_name*/IdentityMinder.ear/library:*path_to_ETPKI_lib_directory*:$LD_LIBRARY_PATH

   Example:

   LD_LIBRARY_PATH=/opt/IBM/WebSphere/AppServer/profiles/AppSrv04/installedApps/bnr210-dNode03Cell/IdentityMinder.ear/library:/opt/CA/SharedComponents/ETPKI/ETPKI/lib:$LD_LIBRARY_PATH

2. Restart the WebSphere application server.

The intent of the FSS Administrative UI is to let you manage SOA Security Manager eTrust SiteMinder FSS. If you are familiar with previous versions of the SiteMinder Policy Server User Interface, you will notice that all SOA Security Manager objects appear in the FSS Administrative UI, except the application objects for Enterprise Policy Management (EPM). You should *not* use the FSS Administrative UI to manage these objects. If you need information while using the FSS Administrative UI, please consult the online help.

Directory Server and Database Requirements

The following sections detail the minimum directory server and database requirements for installing the Administrative UI.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Administrative UI Requirements

Ensure that you meet the following directory server and/or database requirements before Installing the Administrative UI:

- **Object store**—The Administrative UI is an asynchronous application that is event and task-based. The object store stores this information. A supported version of either MS SQL Server or Oracle is required to function as an object store.

  Important! If you are using WebSphere as an application server and MS SQL Server 2005 for the object store, enable XA transactions in MS SQL Server 2005 before installing the Administrative UI or the installation fails. More information exists in How to Enable XA on MS SQL 2005 (see page 351).

- **Administrator user store**—The Administrative UI authenticates SOA Security Manager administrator accounts using an administrator user store. A supported version of either an LDAP directory server or a relational database is required to function as the administrator user store.

Note: All of your administrator accounts must be stored in a single administrator user store. If you are upgrading to r12.1, you may use an existing user store as an administrator store.

Chapter 4: Installing the Policy Server on Windows Systems

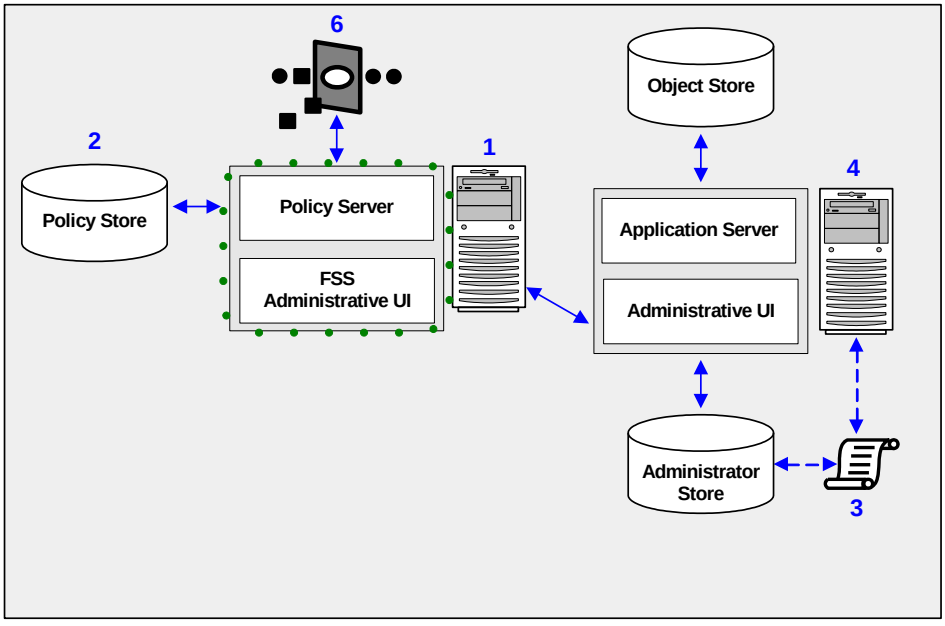
This section contains the following topics:

Installation Road Map

The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- You should have confirmed that the system that is to host the Policy Server meets the minimum system requirements. We recommend doing so before installing the Policy Server.
- The component surrounded by a green dotted line represents the Policy Server, which you install at this point in the installation process. The Policy Server provides policy management, authentication, authorization, and accounting services.

  Note: The FSS Administrative UI is installed with the Policy Server and is only used to manage eTrust SiteMinder FSS. Unless you need to generate WS-Security SAML assertion tokens, use of the FSS Administrative UI is not required. Although installed with the Policy Server, the FSS Administrative UI must be registered with the Policy Server before it may be used. Registering the FSS Administrative UI requires the use of the Administrative UI. Therefore, you must install and configure the Administrative UI before registering the FSS Administrative UI. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services Administrative UI.



The following sections detail how to install the Policy Server.

Before You Install the Policy Server

Be aware of the following before installing the Policy Server:

- **Administrator privileges**—You must be logged into a Windows account with local administrator privileges to install the Policy Server.
- **System path length**—The Policy Server installation fails if the system path length exceeds 1024 characters, including or excluding the SiteMinder added directories.

  Note: We recommend trimming the pre-SOA Security Manager system path to approximately 700 characters for best results.
- **Web Server instance**—Be sure that the Sun Java System or IIS Web server instance is stopped. Stopping the Web server lets the Policy Server installer configure the FSS Administrative UI to operate with the selected Web server instance.
- **Environment variables**—The Policy Server and documentation installations each modify environment variables.
- **IBM Directory Server only**—Using an IBM Directory Servers in your SOA Security Manager environment requires that you edit the V3.matchingrules file by adding the following line:

  MatchingRules=(2.5.13.15 NAME 'integerOrderingMatch' SYNTAX 1.3.6.1.4.1.1466.115.121.1.27)

  The Directory store will not be configured correctly and the necessary SiteMinder objects for the policy store cannot be created if the V3.matchingrules file does not contain the change.

How to Install the Policy Server

To install the Policy Server complete the following procedures:

1. Review the Policy Server component considerations.
2. Review the policy store considerations.
3. Review the FIPS considerations.
4. Gather information for the Policy Server installer.
5. Run the Policy Server installer.
6. Verify the Policy Server installation.
7. (Optional) Enable SNMP event trapping.

8. (Optional) Configure the policy store.

   Note: A SOA Security Manager environment must contain at least one policy store. This step is optional only if you plan on using the Policy Server installer to automatically configure ADAM or a Sun Java System Directory Server as the policy store. Otherwise, you must configure a policy store in a supported LDAP directory server or relational database.

More information:

Reinstall the Policy Server (see page 49)

Policy Server Component Considerations

The Policy Server installer can configure the following components. Review the following before running the Policy Server installer:

- **Federation Security Services Administrative UI**—The FSS Administrative UI is installed with the Policy Server and is for managing eTrust SiteMinder FSS. Use of the FSS Administrative UI is only required If you need to generate WS-Security SAML assertion tokens or are otherwise federating with a partner organization. Although part of the core Policy Server installation, the FSS Administrative UI must be registered with the Policy Server before it may be used. Registering the FSS Administrative UI requires the use of the Administrative UI. Therefore, you must install and configure the Administrative UI before registering the FSS Administrative UI.

  Note: More information on registering the FSS Administrative UI exists in Registering the Federation Security Services Administrative UI.

- **Web Server**—A supported Web server is required to configure the FSS Administrative UI. The Policy Server installer configures the FSS Administrative UI with the selected Web server.

- **OneView Monitor**—The OneView Monitor enables the monitoring of SiteMinder components.

  Note: To use the OneView Monitor, you must have the supported Java SDK and ServletExec ISAPI Windows/IIS installed.

- **SNMP**—Ensure you have an SNMP Service (Master OS Agent) installed with your Windows operating system prior to installing the Policy Server.

  Note: More information on installing the SNMP service exists in the Windows Help system.

- **Policy Store**—The policy store is the repository for Policy Server objects and policy information.

- **Key Database (smkeydatabase)**—The smkeydatabase is a key store used for signing, verification, encryption, and decryption of signed messages with WS-Security tokens, or to produce or consume XML encrypted messages with WS-Security tokens.

  If you choose to configure the smkeydatabase during installation, you are prompted to install the default certificate authority (CA) certificates. You may add additional certificates and private keys to an smkeydatabase after installation.

  Note: More information on the role of the smkeydatabase exists in the *Policy Configuration Guide*.

- **Audit Logs**—You can store audit logs in either a relational database or a text file. After you install the Policy Server, audit logging is set to a text file and not to ODBC by default.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

FIPS Considerations

The Policy Server uses certified Federal Information Processing Standard (FIPS) 140-2 compliant cryptographic libraries.

You must install the SOA Security Manager Policy Server in the default *FIPS-compatibility mode*. In FIPS-compatibility mode, the environment uses existing SOA Security Manager algorithms to encrypt sensitive data on the Policy Server.

Important!: SOA Security Manager does *not* support FIPS-migration mode or FIPS-only mode.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

Gather Information for the Installer

The Policy Server installer requires specific information to install the Policy Server and any optional components.

Note: Installation worksheets are provided to help you gather and record information prior to installing or configuring Policy Server components using the Policy Server Installation Wizard or the Policy Server Configuration Wizard. You may want to print these worksheets and use them to record required information prior to running either wizard.

Required Information

Gather the following required information before running the Policy Server installer or the Configuration wizard. You can use the Required Information Worksheet to record your values.

- **JRE location** - Identify the folder in which the installer can locate the supported JRE and ensure that the JAVA_HOME system variable is set to the correct location. The installer cannot locate the JRE if the JAVA_HOME system variable is incorrectly set.
- **Policy Server installation location** - Determine where the installer should install the Policy Server.

  Default: C:\Program Files\CA
- **Encryption key value** - Determine the encryption key value. An *encryption key* is a case-sensitive, alphanumeric key that secures data sent between the Policy Server and the policy store. All Policy Servers that share a policy store must be configured using the same encryption key. For stronger protection, define a long encryption key.

  Limits: 6 to 24 characters.

More information:

Required Information Worksheet (see page 323)

SiteMinder Key Database Information

You only have to gather SOA Security Manager key database (smkeydatabase) information if you:

- Plan on signing and/or validating signed messages with WS-Security tokens, or producing or consuming XML encrypted messages with WS-Security tokens
- Plan on using features related to eTrust SiteMinder FSS.

The Policy Server installer requires that you enter a password when configuring the smkeydatabase. The smkeydatabase password is used to encrypt the key and certificate data in the key database. You can use the SOA Security Manager Key Database Information Worksheet to record your value.

OneView Monitor Information

You only have to gather OneView Monitor information if you plan on configuring the OneView Monitor.

Gather the following required information to configure the OneView Monitor. You can use the OneView Monitor Information Worksheet to record your values.

- **JDK path**—Identify the path to the required JDK version.
- **ServletExec installation directory**—Identify ServletExec installation directory.

  Example: /usr/local/NewAtlanta/ServletExecAS
- **ServletExec port number**—Determine the port number for the ServletExec instance.
- **Sun Java System administrator directory**—Determine the following information:
  - The installed location of the Sun Java System.
  - The installed location of the Sun Java System Web servers.

  Example: */sunjavasystem_home/location*

  sunjavasystem home

  Specifies the installed location of the Sun Java System.

  location

  Specifies the installed location of the Sun Java System Web servers.
- **Multiple ServletExec instances**—If you have multiple ServletExec instances, determine the instance to which you want to configure the OneView Monitor GUI.

More information:

OneView Monitor Information Worksheet (see page 323)

ADAM Server Information

You only have to gather ADAM server information if you plan on using an ADAM server as a policy store.

Gather the following required information to configure an ADAM server as a policy store. You can use the ADAM Server Information Worksheet to record your values.

- **System IP address**—Identify the IP address of the system where the ADAM server is installed.
- **Server instance port number** - Determine the port number for the ADAM server instance.
- **Root DN of the application partition**—Identify the existing root DN location of the application partition in the ADAM server where the policy store schema data should be installed.

  Example: dc=ca,dc=com
- **ADAM administrator domain name**—Identify the full domain name, including the guid value, of the ADAM administrator.

  Example: CN=user1,CN=people,CN=Configuration,CN=<guid>
- **ADAM administrator password**—Identify the password for the ADAM administrator.
- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Identify the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.
- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

ADAM Server Information Worksheet (see page 324)

Sun Java System Directory Server Information

You only have to gather Sun Java System Directory Server information if you plan on configuring a Sun Java System Directory Server as a policy store.

Gather the following required information to configure a Sun Java System Directory Server as a policy store. You can use the Sun Java System Directory Server Information Worksheet to record your values.

- **System IP address**—Determine the IP address of the system where the Sun Java Systems Directory Server is installed.
- **Directory instance port number**—Determine the port number for the Sun Java Systems Directory Server instance.

  Default: 389
- **Root DN—**Identify the root DN of the Sun Java System Directory Server.

  Example: o=yourorg.com
- **Administrator account**—Identify the user name (Bind DN) for the LDAP administrator account.

  Example: cn=Directory Manager
- **Administrator password**—Identify the password for the Sun Java System Directory Server administrator.
- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Determine the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.
- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

Sun Java System Directory Server Information Worksheet (see page 324)

Run the Installer to Install the Policy Server

You run the SOA Security Manager installer (ca-soasm-12.1-cr001-win32.exe) to install the Policy Server. The executable can be downloaded from the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Note: Before installing the Policy Server, ensure that the system meets the windows requirements.

To run the SOA Security Manager installer to install the Policy Server

1. Exit all applications that are running.
2. Navigate to where the installation executable is located.

3. Double-click ca-soasm-12.1-cr001-win32.exe.

   The SOA Security Manager installation wizard starts.

4. Use the gathered system and component information to install the Policy Server and configure Policy Server components. Consider the following when running the installer:

   - When prompted to select features to install, select CA SOA Security Manager Policy Server.
   - When prompted to select the components you want configured:
     - If you plan on using a relational database or an LDAP directory server other than ADAM or Sun Java System Directory Server as a policy store, deselect the Policy Store option. You manually configure a relational database or any other supported LDAP directory server after installing the Policy Server.
     - If you plan on using WS-Security SAML tokens ensure that the Create SM Key Database/Change SM Key Database Password options are selected.

     Note: If you choose to create a key database, you are prompted to install the default CA certificates. Ensure the Import default CA certificates options is selected and install these certificates. You may add additional certificates and private keys to an smkeydatabase after installation. More information on the role of smkeydatabase exists in the *Policy Configuration Guide*.
   - If you are initializing a policy store, you are prompted to enter a password for the default SOA Security Manager user account. The default account name is SOA Security Manager. This account:
     - is the default Super User account for the FSS Administrative UI. This is not the administrator account for the Administrative UI. You identify a separate administrator account when installing the Administrative UI.
     - is used for all tasks that do not require direct access to the Administrative UI.
   - If you are using IPv6 addresses, ensure your entries include brackets.

     Example: [2001:db8::1428:57ab]
   - When prompted to initialize the LDAP instance do so only to configure a new policy store instance.
   - If you cut and paste path information into the wizard, you must also enter a character before the Next button is enabled.

5. Review the installation settings and proceed.

   The Policy Server and any selected components are installed and configured.

   Note: The FSS Administrative UI was installed during the Policy Server upgrade. The FSS Administrative UI is for managing eTrust SiteMinder FSS. Register the FSS Administrative UI with the Policy Sever after upgrading the policy store. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services UI.

If you experience problems during the installation, you can locate the CA_SOA_Security_Manager_r12.1_InstallLog.log file in *soa_home*\install_config_info

You can also use the ca-ps-details.log file located in *soa_home*\siteminder\install_config_info to check the status of the installer's auto-configuration of an ADAM or Sun Java System Directory Server policy store.

soa_home

Specifies the path to where the SOA Security Manager is installed.

Troubleshoot the Policy Server Installation

Use the following files to troubleshoot the Policy Server installation:

- CA_SiteMinder_Policy_Server_*release*_InstallLog.log

  The installation log contains a summary section that lists the number of successes, warnings, non–fatal errors, and errors that occurred during the installation. Individual installation actions are listed with the respective status.

  release

  Specifies the Policy Server release.

  Location: *siteminder_home*\siteminder\install_config_info

- ca-ps-details.log

  The policy store log details the policy store status.

  Location: *siteminder_home*\siteminder\install_config_info

- smps.log

  The smps.log is created when you start the Policy Server. This log contains the following line if the Policy Server installed successfully:

  [Info] Journaling thread started, will delete commands older than 60 minutes.

  Location: *siteminder_home*\siteminder\log

  siteminder_home

  Specifies the Policy Server installation path.

Enable SNMP Event Trapping

This is an optional step. You only have to enable SNMP trapping if you configured this feature when installing the Policy Server.

Note: Before completing this procedure, ensure you have an SNMP Service installed on the Windows systems.

To enable SNMP event trapping, use the XPSConfig utility to set the event handler library (eventsnmp.dll) to the XPSAudit list. The default location of eventsnmp.dll is *policy_server_home*\bin.

policy_server_home

Specifies the Policy Server installation location.

Note: More information on using the XPSConfig utility to set event handler libraries exists in the *Policy Server Administration Guide*.

To finish configuring SNMP event trapping, configure the snmptrap.conf file. The necessary SNMP prerequisites and procedures are detailed in SNMP Support.

More information:

SNMP Support Overview (see page 315)

Configure a Policy Store

If you did not use the Policy Server installer to configure a policy store automatically, manually configure a supported LDAP directory server or relational database as a policy store.

More information:

Relational Databases as a Policy or Key Store (see page 131)

Install a Policy Server Using the Unattended Installer

After you have installed one or more SOA Security Manager components on one machine, you can reinstall those components on the same machine or install them with the same options on another machine using an unattended installation mode. An unattended installation lets you install or uninstall SOA Security Manager components without any user interaction

The unattended installation uses the ca-soasmr12-installer.properties file generated during the initial install from the information you specified to define the necessary installation parameters, passwords, paths, and so on.

The ca-soasmr12-installer.properties file is located in:
SOA_HOME\install_config_info

SOA_HOME

Specifies the SOA Security Manager installation location. By default, this is:

- Windows: C:\Program Files\CA\SOA Security Manager
- UNIX: ~/CA/SOA_Security_Manager

To run the installer in the unattended installation mode

1. From a system where SOA Security Manager is already installed, copy the ca-soasmr12-installer.properties to a local directory on your system.
2. Download the SOA Security Manager distribution to a temporary location from the Technical Support site:
   - Windows: soasm-r12.1-cr001-win32.zip
   - UNIX: soasm-r12.1-cr001-*os_version*.zip

   os_version

   Specifies sol or linux.
3. Extract the Zip archive into the same local directory as the ca-soasmr12-installer.properties file.
4. Open a console window and navigate to the location where you copied the files.

5. Run the appropriate command for your operating system.

   Windows:

   ca-soasm-12.1-cr001-win32.exe -f ca-soasmr12-installer.properties
   -i silent

   UNIX:

   ca-soasm-12.1-cr001-*os_version*.bin -f ca-soasmr12-installer.properties
   -i silent

   When running this command, if the ca-soasmr12-installer.properties file is not in the same directory as the installation program, make sure you use double quotes if the argument contains spaces.

   For example, on Windows:

   ca-soasm-12.1-cr001-win32.exe -f "C:\Program Files\CA\SOA Security
   Manager\install_config_info\ca-soasmr12-installer.properties" -i silent

   The -i silent setting instructs the installer to run in the unattended installation mode.

   An InstallAnywhere status bar appears, which shows that the unattended SOA Security Manager installer has begun. The installer uses the parameters specified in the ca-soasmr12-installer.properties file.

To stop the installation manually, follow the instructions for your platform:

Windows: Open the Windows Task Manager and stop the ca-soasm-12.1-cr001-win32.exe process.

UNIX: Type Ctrl+C.

To check if the unattended installation completed successfully, see the ca-soasmr12_InstallLog.log file in the *soasm_installation*/install_config_info directory. This log file contains the results of the installation.

Policy Server Configuration Wizard

You use the Policy Server Configuration wizard to configure or reconfigure the following individual Policy Server components after installing the Policy Server:

- FSS Administrative UI
- OneView Monitor GUI
- SNMP support
- SOA Security Manager key database

- An ADAM server policy store
- A Sun Java Systems Directory Server policy store

Note: You cannot change the Policy Server's FIPS mode of operation using the Policy Server Configuration Wizard. More information on changing a Policy Server's FIPS mode of operation exists in the *Upgrade Guide*.

Important! If you already have one Sun Java System Web server instance configured for the OneView Monitor GUI or SNMP do not configure new instances using the Policy Server Configuration wizard. Running the wizard to configure new Web server instances can cause the existing configured Web server instance to fail.

How to Use the Configuration Wizard

Complete the following procedures to use the Policy Server Configuration wizard:

1. Review the Policy Store Considerations.
2. Gather information for the wizard if you are configuring:
   - The SOA Security Manager key database (smkeydatabase)
   - The OneView Monitor
   - An ADAM server policy store
   - A Sun Java System Directory Server policy store
3. Run the Policy Server Configuration wizard.

Policy Store Considerations

Consider the following before running the Policy Server installer or the Policy Server Configuration wizard:

- The Policy Server installer and configuration wizard can automatically configure the policy store in an Active Directory Application Mode (ADAM) or a Sun Java System Directory Server.

  Note: Ensure you have met the prerequisites for configuring ADAM as a policy store before running the installation.

- You may use any other supported LDAP or relational database as a policy store, but must manually configure the policy store after installing the Policy Server. Manually configuring an LDAP directory server or relational database as a policy store is detailed in this guide.

Note: The Policy Server installer and configuration wizard cannot automatically configure a policy store that is being connected to using an SSL connection.

More information:

ADAM Server Prerequisites (see page 121)

Gather Information for the Configuration Wizard

The Policy Server Configuration Wizard requires specific information to configure Policy Server components.

Note: Installation worksheets are provided to help you gather and record information prior to installing or configuring Policy Server components using the Policy Server Installation Wizard or the Policy Server Configuration Wizard. You may want to print these worksheets and use them to record required information prior to running either wizard.

SiteMinder Key Database Information

You only have to gather SOA Security Manager key database (smkeydatabase) information if you:

- Plan on using features related to eTrust SiteMinder FSS.
- Plan on configuring a SOA Security Manager Information Card Authentication scheme, for example, for the support of Microsoft CardSpace.

The Policy Server installer requires that you enter a password when configuring the smkeydatabase. The smkeydatabase password is used to encrypt the key and certificate data in the key database. You can use the SOA Security Manager Key Database Information Worksheet to record your value.

OneView Monitor Information

You only have to gather OneView Monitor information if you plan on configuring the OneView Monitor.

Gather the following required information to configure the OneView Monitor. You can use the OneView Monitor Information Worksheet to record your values.

- **JDK path**—Identify the path to the required JDK version.
- **ServletExec installation directory**—Identify ServletExec installation directory.

  Example: /usr/local/NewAtlanta/ServletExecAS

- **ServletExec port number**—Determine the port number for the ServletExec instance.
- **Sun Java System administrator directory**—Determine the following information:
  - The installed location of the Sun Java System.
  - The installed location of the Sun Java System Web servers.

  Example: */sunjavasystem_home/location*

  sunjavasystem home

  Specifies the installed location of the Sun Java System.

  location

  Specifies the installed location of the Sun Java System Web servers.
- **Multiple ServletExec instances**—If you have multiple ServletExec instances, determine the instance to which you want to configure the OneView Monitor GUI.

More information:

OneView Monitor Information Worksheet (see page 323)

ADAM Server Information

You only have to gather ADAM server information if you plan on using an ADAM server as a policy store.

Gather the following required information to configure an ADAM server as a policy store. You can use the ADAM Server Information Worksheet to record your values.

- **System IP address**—Identify the IP address of the system where the ADAM server is installed.
- **Server instance port number** - Determine the port number for the ADAM server instance.
- **Root DN of the application partition**—Identify the existing root DN location of the application partition in the ADAM server where the policy store schema data should be installed.

  Example: dc=ca,dc=com
- **ADAM administrator domain name**—Identify the full domain name, including the guid value, of the ADAM administrator.

  Example: CN=user1,CN=people,CN=Configuration,CN=<guid>
- **ADAM administrator password**—Identify the password for the ADAM administrator.

- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Identify the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.

- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

ADAM Server Information Worksheet (see page 324)

Sun Java System Directory Server Information

You only have to gather Sun Java System Directory Server information if you plan on configuring a Sun Java System Directory Server as a policy store.

Gather the following required information to configure a Sun Java System Directory Server as a policy store. You can use the Sun Java System Directory Server Information Worksheet to record your values.

- **System IP address**—Determine the IP address of the system where the Sun Java Systems Directory Server is installed.
- **Directory instance port number**—Determine the port number for the Sun Java Systems Directory Server instance.

  Default: 389
- **Root DN—**Identify the root DN of the Sun Java System Directory Server.

  Example: o=yourorg.com
- **Administrator account**—Identify the user name (Bind DN) for the LDAP administrator account.

  Example: cn=Directory Manager
- **Administrator password**—Identify the password for the Sun Java System Directory Server administrator.

- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Determine the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.

- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

Sun Java System Directory Server Information Worksheet (see page 324)

Run the Configuration Wizard

To run the configuration wizard

1. Exit all applications that are running.
2. Navigate to *siteminder_home*\siteminder\install_config_info and double-click ca-ps-config.exe.

   The Policy Server configuration wizard starts.

   siteminder_home

   Specifies the Policy Server installation path.
3. Use the system and component information you have gathered to configure a policy store and individual components.

   Note: When prompted to initialize the LDAP instance do so only to configure a new policy store instance.

4. Review the installation settings and click Install.

   The wizard configures the selected components to work with the Policy Server.

   Note: This can take several minutes.

5. Click Done and reboot the system.

   The components you selected are configured.

Note: If you experience problems, you can locate the Policy Server installation log file and the policy store details file in *siteminder_home*\siteminder\install_config_info.

siteminder_home

Specifies the Policy Server installation path.

More information:

Troubleshoot the Policy Server Installation (see page 40)

Reinstall the Policy Server

Reinstalling the Policy Server over an existing Policy Server of the same version lets you restore lost application files or restore the Policy Server's default installation settings.

To reinstall the Policy Server

1. Stop the Policy Server using the Policy Server Management Console.

   Note: More information on stopping and starting the Policy Server exists in the *Policy Server Administration Guide*.

2. Close the Policy Server Management Console.
3. Install the Policy Server.
4. Start the Policy Server using the Policy Server Management Console.

More information:

How to Install the Policy Server (see page 31)

How to Uninstall the Policy Server

To uninstall the Policy Server complete the following procedures:

1. Set the JRE in the Path System Variable.
2. Shut down all instances of the Policy Server Management Console.
3. Remove Policy Server References from Agent Host Files.
4. Uninstall the Policy Server.

Set the JRE in the Path Variable

You set the JRE in path variable when uninstalling the Policy Server, Web Agent and SDK to prevent the uninstallation program from stopping and issuing one of the following error messages:

- "Could not find a valid Java virtual machine to load. You need to reinstall a supported Java virtual machine."
- "No Java virtual machine could be found from your PATH environment variable. You must install a VM prior to running this program."

To Set the JRE in the Path variable

1. Open the Windows Control Panel.
2. Double-click System.
3. Add the location of the JRE to the Path system variable in the Environment Variables dialog.

Remove Policy Server References from Agent Host Files

You remove the Policy Server reference from the SmHost.conf file to prevent unexpected results from the SOA Agent once the Policy Server is uninstalled.

To remove the Policy Server reference

1. Navigate to *soa_agent_home*/config.

   soa_agent_home

   Specifies the installation directory of the SOA Agent.

2. Open the SmHost.conf file in a text editor.

3. Delete the line that begins with "policyserver=".

   Note: This line contains the IP address and port numbers for the Policy Server you are uninstalling.

4. Save SmHost.conf.

   The SmHost.conf file no longer references the Policy Server you are uninstalling.

Uninstall the Policy Server

To uninstall the Policy Server when it is no longer required on the system, run the SOA Security Manager uninstall wizard.

To uninstall SOA Security Manager components on Windows or UNIX systems

1. Navigate to the *SOA_HOME*\install_config_info (Windows) or *SOA_HOME*/install_config_info (UNIX) directory and run the SOA Security Manager uninstall wizard to remove core SOA Security Manager components:

   - Windows: soa-uninstall.cmd
   - UNIX: soa-uninstall.sh

   SOA_HOME

   Specifies the SOA Security Manager installation location.

   The uninstall wizard starts.

2. Choose whether you want to perform a complete uninstall or whether to uninstall specific features and proceed.
3. If you chose to uninstall only specific features, select the installed components that you want to uninstall and proceed.

The uninstall wizard removes all selected SOA Security Manager components.

1.

2.

3.

4.

Remove Leftover Items

Manually remove the following folders, registry settings, and virtual directories after uninstalling the Policy Server to keep the system as clean as possible.

- **Windows system**
  - *Siteminder_home*\bin
  - *Siteminder_home*\install_config_info

  Note: If you are planning on reinstalling the Policy Server, ensure these folders are removed.
- **AdventNet software registry entry** - Delete the AdventNet software registry entry only if the software was not on the system prior to installing the Policy Server. You can find the registry entry under Registry entries under HKEY_LOCAL_MACHINE\SOFTWARE\Advent,Inc.
- **IIS virtual directories** - Delete the following virtual directories using the IIS Microsoft Management Console
  - 'SiteMinder'
  - 'SiteMinderCgi'
  - 'SiteMinderMonitor'
  - 'netegrity_docs'

Remove Leftover Services

After uninstalling the Policy Server and rebooting the machine, the following services may not be removed.:

- SiteMinder Health Monitor Service
- SiteMinder Policy Server
- SNMP Agent

To manually remove leftover services

1. Stop each service.
2. Remove the following Windows registry key, as necessary:

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<Registry_Key_Name>

Registry_Key_Name

Specifies the registry key name of the service that is to be removed:

- SMServMon (SiteMinder Health Monitor Service)
- SMPolicySrv (SiteMinder Policy Server)
- Agent Service (SNMP Agent)

The leftover services are removed.

Chapter 5: Installing the Policy Server on UNIX Systems

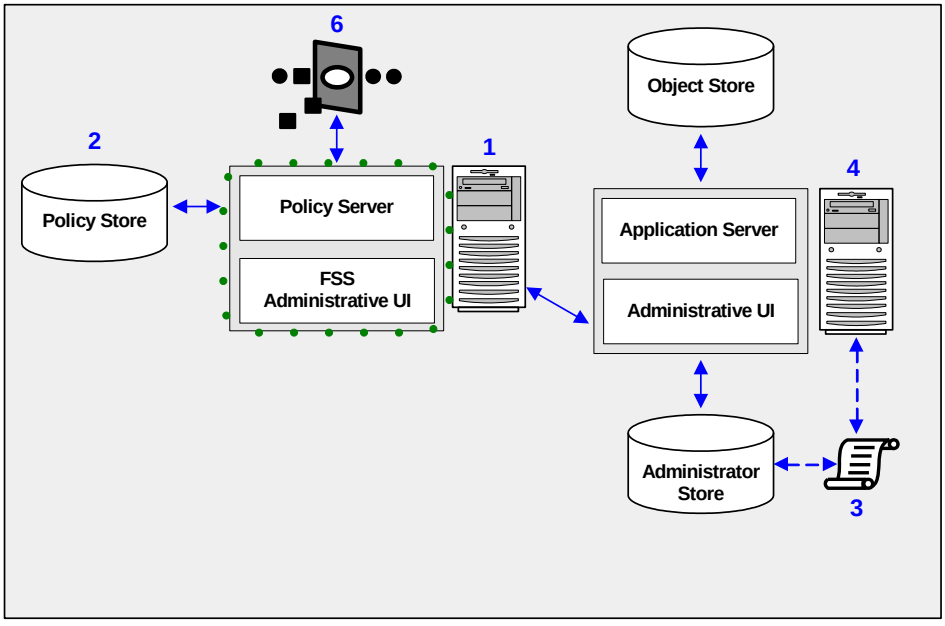
This section contains the following topics:

Installation Road Map

The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- You should have confirmed that the system that is to host the Policy Server meets the minimum system requirements. We recommend doing so before installing the Policy Server.
- The component surrounded by a green dotted line represents the Policy Server, which you install at this point in the installation process. The Policy Server provides policy management, authentication, authorization, and accounting services.

  Note: The FSS Administrative UI is installed with the Policy Server and is only used to manage eTrust SiteMinder FSS. Unless you need to generate WS-Security SAML assertion tokens, use of the FSS Administrative UI is not required. Although installed with the Policy Server, the FSS Administrative UI must be registered with the Policy Server before it may be used. Registering the FSS Administrative UI requires the use of the Administrative UI. Therefore, you must install and configure the Administrative UI before registering the FSS Administrative UI. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services Administrative UI.



The following sections detail how to install the Policy Server.

Support Considerations for Solaris 10

Consider the following when planning to run one or more Policy Servers in a Solaris 10 environment:

- Running a Policy Server in either the global zone or the whole-root non-global zone is supported. Whole-root zones increase configuration flexibility, but increase resource usage because they do not share directories.

  Note: The Policy Server is not supported on Sparse-root zones.

- Running multiple Policy Server instances, on multiple whole-root zones is supported. However, consider that only one Policy Server instance is supported in each whole-root or global zone.

Solaris and HP-UX Patches

For a list of required and recommended Solaris and HP-UX patches, see the *Policy Server Release Notes*.

How to Prepare for the Policy Server Installation

Before you install the Policy Server on a UNIX system, complete the following steps, if applicable:

1. Create a New UNIX Account.
2. Modify the UNIX System Parameters.
3. Unset the Localization Variables.
4. Unset the LANG Environment Variable.

Create a New UNIX Account

Create a new UNIX account named smuser with the default shell as ksh. You may also need to modify the profile for the smuser account, as indicated later in this chapter.

Important! You should install the Policy Server using the smuser UNIX account, but do not configure the Sun Java System or Apache on Linux Web Server for the FSS Administrative UI or the OneView Monitor GUI because the installer modifies the Web server's configuration files and smuser does not have the appropriate root privileges. Thus, when you run the Policy Server installer, do not select Web Server(s) or OneView Monitor when prompted to choose components.

After the Policy Server installation is complete, run the Policy Server Configuration Wizard (located in *siteminder_installation*\install_config_info\ca-ps-config.bin) as root to configure the FSS Administrative UI or the OneView Monitor GUI.

Modify the UNIX System Parameters

When the Policy Server is placed under load, it opens a large number of sockets and files. This can become a problem if the default limit parameters are not appropriate for the load. You modify the default limit parameters to avoid associated problems.

To view the default limit parameters, type ulimit -a. The system displays a message similar to the following:

```
$ ulimit -a
time(seconds)                   unlimited
file(blocks)                    unlimited
data(kbytes                     2097148
stack(kbytes)                   8192
coredump(blocks)                unlimited
nofiles(descriptors)            256
vmemory(kbytes)                 unlimited
```

The nofiles parameter is set to 256 in this example. This is the total number of files (sockets + files descriptors) that this shell and its descendants have been allocated. If this parameter is not set high enough, the Policy Server returns numerous socket errors. The most common socket error is 10024, or too many open files.

You must increase this parameter value for proper Policy Server operation under load. You can change this value by running the ulimit -n command. For example, to set nofiles to 1024, place the ulimit -n 1024 command in the .profile or smprofile.ksh of the smuser account. The Policy Server is bound by the nofiles parameter within smuser's ulimit for the number of connections to it.

Note: On HP-UX systems, prior to the Policy Server installation, check your .profile file for a set +u option. If it has a set -u option, do a set +u to nullify it. A setting of set -u will cause a problem when the installation sets a SHLIB_PATH for smuser.

Unset Localization Variables

The LC_* variables are sometimes set by default in the .profile or smprofile.ksk files of the smuser account. Use of the LC_* environment variables are not permitted and use must unset them before installing the Policy Server.

To unset the LC_* environment variables open the .profile or smprofile.ksk files of the smuser account and unset them.

Unset the LANG Environment Variable

The LANG environment variable are not permitted and you must unset it before installing the Policy Server.

To unset the variable, add the unset LANG command to the smprofile.ksh file.

Before You Install the Policy Server

Consider the following before installing the Policy Server:

- **Administrator privileges**—Ensure you have created a new UNIX account and understand which components require root administrator privileges.

  Note: More information on the required administrator privileges exists in How to Prepare for the Policy Server Installation.

- **Free space in /tmp**—The Policy Server install requires 200 MB of free space in /tmp.

- **System path length**—The Policy Server installation fails if the system path length exceeds 1024 characters, including or excluding the SiteMinder added directories.

  Note: We recommend trimming the pre-SOA Security Manager system path to approximately 700 characters for best results.

- **Web Server instance**—Ensure that the Sun Java System Web server instance is stopped before installing the Policy Server.

- **Telnet or other terminal emulation software**—If you plan on installing the Policy Server using Telnet or other terminal emulation software, you should install using a console window. The installer throws a Java exception and exits if you try to run a GUI through a telnet window.

- **Exceed X-windows application**—Running the Policy Server installer or the Policy Server Configuration Wizard (ca-ps-config.bin) using an Exceed X-windows application may cause text in the dialog box to truncate due to unavailable fonts using Exceed. This limitation has no affect on the Policy Server installation or configuration.
- **Environment variables**—The Policy Server and documentation installations each modify environment variables.
- **IBM Directory Server only**—Using an IBM Directory Servers in your SOA Security Manager environment requires that you edit the V3.matchingrules file by adding the following line:

  MatchingRules=(2.5.13.15 NAME 'integerOrderingMatch' SYNTAX 1.3.6.1.4.1.1466.115.121.1.27)

  The Directory store will not be configured correctly and the necessary SiteMinder policy store objects cannot be created if the V3.matchingrules file does not contain the change.

How to Install the Policy Server

To install the Policy Server, complete the following steps:

1. Review the Policy Server component considerations.
2. Review the Policy Store considerations.
3. Review the FIPS considerations.
4. Gather information for the Policy Server installer.
5. Run the Policy Server installer.
6. Verify the Policy Server installation.
7. (Optional) Restart the SNMP daemon.
8. (Optional) Configure the policy store.

   Note: A SOA Security Manager environment must contain at least one policy store. This step is optional only if you plan on using the Policy Server installer to automatically configure an ADAM or Sun One LDAP directory server as the policy store. Otherwise, you must configure a policy store in a supported LDAP directory server or relational database.

Policy Store Considerations

Consider the following before running the Policy Server installer or the Policy Server Configuration wizard:

- The Policy Server installer and configuration wizard can automatically configure the policy store in an Active Directory Application Mode (ADAM) or a Sun Java System Directory Server.

  Note: Ensure you have met the prerequisites for configuring ADAM as a policy store before running the installation.

- You may use any other supported LDAP or relational database as a policy store, but must manually configure the policy store after installing the Policy Server. Manually configuring an LDAP directory server or relational database as a policy store is detailed in this guide.

Note: The Policy Server installer and configuration wizard cannot automatically configure a policy store that is being connected to using an SSL connection.

More information:

ADAM Server Prerequisites (see page 121)

Policy Server Component Considerations

The Policy Server installer can configure the following components. Consider the following before running the Policy Server installer:

- **FSS Administrative UI**—The FSS Administrative UI is installed with the Policy Server and is for managing eTrust SiteMinder FSS. Use of the FSS Administrative UI is only required If you need to generate WS-Security SAML assertion tokens or are otherwise federating with a partner organization. Although part of the core Policy Server installation, the FSS Administrative UI must be registered with the Policy Server before it may be used. Registering the FSS Administrative UI requires the use of the Administrative UI. Therefore, you must install and configure the Administrative UI before registering the FSS Administrative UI.

  Note: More information on registering the FSS Administrative UI exists in Registering the Federation Security Services Administrative UI.

- **Web Server**—A supported Web server is required to configure the FSS Administrative UI. The Policy Server installer configures the FSS Administrative UI with the selected Web server.

- **OneView Monitor**—The OneView Monitor enables the monitoring of SiteMinder components.

  Note: To use the OneView Monitor, you must have the supported Java SDK and ServletExec ISAPI Windows/IIS installed.

- **SNMP**—You must know the root user's password and have a native SunSolstice Master Agent to enable SNMP support.
- **Policy Store**—The policy store is the repository for Policy Server objects and policy information.
- **Key Database (smkeydatabase)**—The smkeydatabase is a key store used for signing, verification, encryption, and decryption of signed messages with WS-Security tokens, or to produce or consume XML encrypted messages with WS-Security tokens.

  If you choose to configure the smkeydatabase during installation, you are prompted to install the default certificate authority (CA) certificates. You may add additional certificates and private keys to an smkeydatabase after installation.

  Note: More information on the role of the smkeydatabase exists in the *Policy Configuration Guide*.
- **Audit Logs**—You can store audit logs in either a relational database or a text file. After you install the Policy Server, audit logging is set to a text file and not to ODBC by default.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

FIPS Considerations

The Policy Server uses certified Federal Information Processing Standard (FIPS) 140-2 compliant cryptographic libraries.

You must install the SOA Security Manager Policy Server in the default *FIPS-compatibility mode*. In FIPS-compatibility mode, the environment uses existing SOA Security Manager algorithms to encrypt sensitive data on the Policy Server.

Important!: SOA Security Manager does *not* support FIPS-migration mode or FIPS-only mode.

Gather Information for the Installer

The Policy Server installer requires specific information to install the Policy Server and any optional components.

Note: Installation worksheets are provided to help you gather and record information prior to installing or configuring Policy Server components using the Policy Server Installation Wizard or the Policy Server Configuration Wizard. You may want to print these worksheets and use them to record required information prior to running either wizard.

Required Information

Gather the following required information before running the Policy Server installer or the Configuration wizard. You can use the Required Information Worksheet to record your values.

- **JRE location**—Identify the folder in which the installer can locate the supported JRE and ensure that the JRE is set correctly in the PATH variable. The installer cannot locate the JRE if the JRE is not set correctly in the PATH variable.
- **Policy Server Installation location**—Determine where the installer should install the Policy Server.
- **Encryption key value**—Determine the encryption key value. An *encryption key* is a case-sensitive, alphanumeric key that secures data sent between the Policy Server and the policy store. All Policy Servers that share a policy store must be configured using the same encryption key. For stronger protection, define a long encryption key.

  Limits: 6 to 24 characters.

More information:

Required Information Worksheet (see page 323)

SiteMinder Key Database Information

You only have to gather SOA Security Manager key database (smkeydatabase) information if you:

- Plan on using features related to eTrust SiteMinder FSS.
- Plan on configuring a SOA Security Manager Information Card Authentication scheme, for example, for the support of Microsoft CardSpace.

The Policy Server installer requires that you enter a password when configuring the smkeydatabase. The smkeydatabase password is used to encrypt the key and certificate data in the key database. You can use the SOA Security Manager Key Database Information Worksheet to record your value.

OneView Monitor Information

You only have to gather OneView Monitor information if you plan on configuring the OneView Monitor.

Gather the following required information to configure the OneView Monitor. You can use the OneView Monitor Information Worksheet to record your values.

- **JDK path**—Identify the path to the required JDK version.
- **ServletExec installation directory**—Identify ServletExec installation directory.

  Example: /usr/local/NewAtlanta/ServletExecAS
- **ServletExec port number**—Determine the port number for the ServletExec instance.
- **Sun Java System administrator directory**—Determine the following information:
  - The installed location of the Sun Java System.
  - The installed location of the Sun Java System Web servers.

  Example: */sunjavasystem_home/location*

  sunjavasystem home

  Specifies the installed location of the Sun Java System.

  location

  Specifies the installed location of the Sun Java System Web servers.
- **Multiple ServletExec instances**—If you have multiple ServletExec instances, determine the instance to which you want to configure the OneView Monitor GUI.

More information:

OneView Monitor Information Worksheet (see page 323)

Sun Java System Directory Server Information

You only have to gather Sun Java System Directory Server information if you plan on configuring a Sun Java System Directory Server as a policy store.

Gather the following required information to configure a Sun Java System Directory Server as a policy store. You can use the Sun Java System Directory Server Information Worksheet to record your values.

- **System IP address**—Determine the IP address of the system where the Sun Java Systems Directory Server is installed.
- **Directory instance port number**—Determine the port number for the Sun Java Systems Directory Server instance.

  Default: 389
- **Root DN—**Identify the root DN of the Sun Java System Directory Server.

  Example: o=yourorg.com
- **Administrator account**—Identify the user name (Bind DN) for the LDAP administrator account.

  Example: cn=Directory Manager
- **Administrator password**—Identify the password for the Sun Java System Directory Server administrator.
- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Determine the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.
- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

Sun Java System Directory Server Information Worksheet (see page 324)

ADAM Server Information

You only have to gather ADAM server information if you plan on using an ADAM server as a policy store.

Gather the following required information to configure an ADAM server as a policy store. You can use the ADAM Server Information Worksheet to record your values.

- **System IP address**—Identify the IP address of the system where the ADAM server is installed.
- **Server instance port number** - Determine the port number for the ADAM server instance.
- **Root DN of the application partition**—Identify the existing root DN location of the application partition in the ADAM server where the policy store schema data should be installed.

  Example: dc=ca,dc=com
- **ADAM administrator domain name**—Identify the full domain name, including the guid value, of the ADAM administrator.

  Example: CN=user1,CN=people,CN=Configuration,CN=<guid>
- **ADAM administrator password**—Identify the password for the ADAM administrator.
- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Identify the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.
- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

ADAM Server Information Worksheet (see page 324)

Run the Installer to Install the Policy Server Using a GUI

You run the respective UNIX installation executable to install the Policy Server. The installation executables are in the SOA Security Manager kit available on the Technical Support site. They are the following:

- **Solaris**—ca-soasm-12.1-cr001-sol.bin
- **Red Hat Linux**—ca-soasm-12.1-cr001-linux.bin

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Note: Extract *all* files and directories in the SOA Security Manager kit to a common location on a local drive and keep them together—the SOA Security Manager installer cannot run without the included supporting files.

If you do not have sufficient rights, add executable permission to the install file by running the following command:

chmod+x ca-soasm-12.1-cr001-*os_version*.bin

os_version

Specifies sol or linux.

Important! If you execute the SOA Security Manager installer across different subnets it can crash. To avoid this problem, install the Policy Server directly on the host system.

Note: Before installing the Policy Server, ensure that the system meets the UNIX requirements and you have gathered the necessary information for the installer.

To run the Policy Server installer using a GUI

1. Exit all applications that are running.
2. Open a command window and navigate to where the install program is located.

3. Enter the following command:

   sh./ca-soasm-12.1-cr001-*os_version*.bin

   The SOA Security Manager installation wizard starts.

4. Use the gathered system and component information to install the Policy Server and configure Policy Server components. Consider the following when running the installer:

   - When prompted to select features to install, select CA SOA Security Manager Policy Server.
   - When prompted to select the components you want configured:
     - If you plan on using a relational database or an LDAP directory server other than ADAM or Sun Java System Directory Server as a policy store, deselect the Policy Store option. You manually configure a relational database or any other supported LDAP directory server after installing the Policy Server.
     - If you plan on using WS-Security SAML tokens ensure that the Create SM Key Database/Change SM Key Database Password options are selected.

     Note: If you choose to create a key database, you are prompted to install the default CA certificates. Ensure the Import default CA certificates options is selected and install these certificates. You may add additional certificates and private keys to an smkeydatabase after installation. More information on the role of smkeydatabase exists in the *Policy Configuration Guide*.
   - If you are initializing a policy store, you are prompted to enter a password for the default SOA Security Manager user account. The default account name is SOA Security Manager. This account:
     - is the default Super User account for the FSS Administrative UI. This is not the administrator account for the Administrative UI. You identify a separate administrator account when installing the Administrative UI.
     - is used for all tasks that do not require direct access to the Administrative UI.
   - If you are using IPv6 addresses, ensure your entries include brackets.

     Example: [2001:db8::1428:57ab]
   - When prompted to initialize the LDAP instance do so only to configure a new policy store instance.
   - If you cut and paste path information into the wizard, you must also enter a character before the Next button is enabled.

5. Review the installation settings and proceed.

   The Policy Server and any selected components are installed and configured.

   Note: The FSS Administrative UI was installed during the Policy Server upgrade. The FSS Administrative UI is for managing eTrust SiteMinder FSS. Register the FSS Administrative UI with the Policy Sever after upgrading the policy store. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services UI.

6. Review the installation settings and click Install.

   The Policy Server and any selected components are installed and configured.

Note: If you require the FSS Administrative UI, register it with the Policy Server after installing and registering the Administrative UI. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services UI.

If you experience problems during an installation or upgrade, you can locate the CA_SOA_Security_Manager_r12.1_InstallLog.log file in *soa_home*/install_config_info/

You can also refer to the ca-ps-details.log file located in *soa_home*/siteminder/install_config_info to check the status of the installer's auto-configuration of an ADAM or Sun Java System Directory Server policy store.

soa_home

Specifies the path to where SOA Security Manager is installed.

Run the Installer to Install the Policy Server Using a UNIX Console

You run the respective UNIX installation executable to install the Policy Server. The installation executables are in the SOA Security Manager kit available on the Technical Support site. They are the following:

- **Solaris**—ca-soasm-12.1-cr001-sol.bin
- **Red Hat Linux**—ca-soasm-12.1-cr001-linux.bin

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.

3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Note: Extract *all* files and directories in the SOA Security Manager kit to a common location on a local drive and keep them together—the SOA Security Manager installer cannot run without the included supporting files.

If you do not have sufficient rights, add executable permission to the install file by running the following command:

chmod+x ca-soasm-12.1-cr001-*os_version*.bin

os_version

Specifies sol or linux.

Important! If you execute the SOA Security Manager installer across different subnets it can crash. To avoid this problem, install the Policy Server directly on the host system.

Note: Before installing the Policy Server, ensure that the system meets the UNIX requirements and you have gathered the necessary information for the installer.

To run the Policy Server installer using a UNIX console

1. Exit all applications that are running.
2. Enter the following command in a UNIX shell:

   sh ./ca-soasm-12.1-cr001-*os_version*.bin -i

   The installation starts.

   Note: When prompted to select from a list of numbered choices, enter the numbers separated by commas (,). To select none of the features, enter only a comma.
3. Use the gathered system and component information to install the Policy Server and configure Policy Server components. Consider the following when running the installer:

   - When prompted to select features to install, select CA SOA Security Manager Policy Server.
   - When prompted to select the components you want configured:
     - If you plan on using a relational database or an LDAP directory server other than ADAM or Sun Java System Directory Server as a policy store, deselect the Policy Store option. You manually configure a relational database or any other supported LDAP directory server after installing the Policy Server.
     - If you plan on using WS-Security SAML tokens ensure that the Create SM Key Database/Change SM Key Database Password options are selected.

     Note: If you choose to create a key database, you are prompted to install the default CA certificates. Ensure the Import default CA certificates options is selected and install these certificates. You may add additional certificates and private keys to an smkeydatabase after installation. More information on the role of smkeydatabase exists in the *Policy Configuration Guide*.
   - If you are initializing a policy store, you are prompted to enter a password for the default SOA Security Manager user account. The default account name is SOA Security Manager. This account:
     - is the default Super User account for the FSS Administrative UI. This is not the administrator account for the Administrative UI. You identify a separate administrator account when installing the Administrative UI.
     - is used for all tasks that do not require direct access to the Administrative UI.

- If you are using IPv6 addresses, ensure your entries include brackets.

  Example: [2001:db8::1428:57ab]

- When prompted to initialize the LDAP instance do so only to configure a new policy store instance.
- If you cut and paste path information into the wizard, you must also enter a character before the Next button is enabled.

4. Review the installation settings and proceed.

   The Policy Server and any selected components are installed and configured.

   Note: The FSS Administrative UI was installed during the Policy Server upgrade. The FSS Administrative UI is for managing eTrust SiteMinder FSS. Register the FSS Administrative UI with the Policy Sever after upgrading the policy store. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services UI.

5. Review the installation settings and press Enter.

   The Policy Server is installed.

   Note: This may take several minutes.

6. Press Enter.

   The installer closes

7. Reboot the system.

   The Policy Server and selected components are installed and configured.

   Note: If you require the FSS Administrative UI, register it with the Policy Server after installing and registering the Administrative UI. More information on registering the FSS Administrative UI exists in Registering the Federation Security Services UI.

If you experience problems during an installation or upgrade, you can locate the CA_SOA_Security_Manager_r12.1_InstallLog.log file in *soa_home*/install_config_info/

You can also refer to the ca-ps-details.log file located in *soa_home*/siteminder/install_config_info to check the status of the installer's auto-configuration of an ADAM or Sun Java System Directory Server policy store.

soa_home

Specifies the path to where SOA Security Manager is installed.

Troubleshoot the Policy Server Installation

Use the following files to troubleshoot the Policy Server installation:

- CA_SiteMinder_Policy_Server_*release*_InstallLog.log

  The installation log contains a summary section that lists the number of successes, warnings, non–fatal errors, and errors that occurred during the installation. Individual installation actions are listed with the respective status.

  release

  Specifies the Policy Server release.

  Location: *siteminder_home*\siteminder\install_config_info

- ca-ps-details.log

  The policy store log details the policy store status.

  Location: *siteminder_home*\siteminder\install_config_info

- smps.log

  The smps.log is created when you start the Policy Server. This log contains the following line if the Policy Server installed successfully:

  [Info] Journaling thread started, will delete commands older than 60 minutes.

  Location: *siteminder_home*\siteminder\log

  siteminder_home

  Specifies the Policy Server installation path.

Restart the SNMP Daemon

You only have to restart the SNMP daemon if you configured SNMP during the Policy Server installation.

To restart the SNMP daemon

1. Enter S76snmpdx stop in /etc/rc3.d.

   The SNMP daemon stops.

2. Enter S76snmpdx start in /etc/rc3.d.

   The SNMP daemon starts.

Configure a Policy Store

If you did not use the Policy Server installer to configure a policy store automatically, manually configure a supported LDAP directory server or relational database as a policy store.

More information:

Relational Databases as a Policy or Key Store (see page 131)

Configure Auto Startup

You configure auto startup to ensure that the Policy Server restarts automatically when the Solaris system is rebooted.

To configure auto startup

1. Modify the S98M script by replacing every instance of the string "nete_ps_root" with an explicit path to the Policy Server installation directory.

   Example: /export/ca/SOA Security Manager/siteminder

2. Change the directory to the siteminder installation directory.
3. Enter **su** and press ENTER**.**

   Note: Do not use the suse command.

   You are prompted for a password.

4. Enter the root password and press ENTER.
5. Enter **$ cp S98sm /etc/rc2.d** and press ENTER.

   s98sm automatically calls the stop-all and start-all executables, which stop and start the Policy Server's service when the Solaris system is rebooted.

Note: If you are using a local LDAP directory server as a policy store, you must configure the LDAP directory to start automatically before starting the Policy Server automatically.

Install a Policy Server Using the Unattended Installer

After you have installed one or more SOA Security Manager components on one machine, you can reinstall those components on the same machine or install them with the same options on another machine using an unattended installation mode. An unattended installation lets you install or uninstall SOA Security Manager components without any user interaction

The unattended installation uses the ca-soasmr12-installer.properties file generated during the initial install from the information you specified to define the necessary installation parameters, passwords, paths, and so on.

The ca-soasmr12-installer.properties file is located in:
SOA_HOME\install_config_info

SOA_HOME

Specifies the SOA Security Manager installation location. By default, this is:

- Windows: C:\Program Files\CA\SOA Security Manager
- UNIX: ~/CA/SOA_Security_Manager

To run the installer in the unattended installation mode

1. From a system where SOA Security Manager is already installed, copy the ca-soasmr12-installer.properties to a local directory on your system.
2. Download the SOA Security Manager distribution to a temporary location from the Technical Support site:
   - Windows: soasm-r12.1-cr001-win32.zip
   - UNIX: soasm-r12.1-cr001-*os_version*.zip

   os_version

   Specifies sol or linux.
3. Extract the Zip archive into the same local directory as the ca-soasmr12-installer.properties file.
4. Open a console window and navigate to the location where you copied the files.

5. Run the appropriate command for your operating system.

   Windows:

   ca-soasm-12.1-cr001-win32.exe -f ca-soasmr12-installer.properties
   -i silent

   UNIX:

   ca-soasm-12.1-cr001-*os_version*.bin -f ca-soasmr12-installer.properties
   -i silent

   When running this command, if the ca-soasmr12-installer.properties file is not in the same directory as the installation program, make sure you use double quotes if the argument contains spaces.

   For example, on Windows:

   ca-soasm-12.1-cr001-win32.exe -f "C:\Program Files\CA\SOA Security
   Manager\install_config_info\ca-soasmr12-installer.properties" -i silent

   The -i silent setting instructs the installer to run in the unattended installation mode.

   An InstallAnywhere status bar appears, which shows that the unattended SOA Security Manager installer has begun. The installer uses the parameters specified in the ca-soasmr12-installer.properties file.

To stop the installation manually, follow the instructions for your platform:

Windows: Open the Windows Task Manager and stop the ca-soasm-12.1-cr001-win32.exe process.

UNIX: Type Ctrl+C.

To check if the unattended installation completed successfully, see the ca-soasmr12_InstallLog.log file in the *soasm_installation*/install_config_info directory. This log file contains the results of the installation.

Policy Server Configuration Wizard

You use the Policy Server Configuration wizard to configure or reconfigure the following individual Policy Server components after installing the Policy Server:

- FSS Administrative UI
- OneView Monitor GUI
- SNMP support
- SOA Security Manager key database

- An ADAM server policy store
- A Sun Java Systems Directory Server policy store

Note: You cannot change the Policy Server's FIPS mode of operation using the Policy Server Configuration Wizard. More information on changing a Policy Server's FIPS mode of operation exists in the *Upgrade Guide*.

Important! If you already have one Sun Java System Web server instance configured for the OneView Monitor GUI or SNMP do not configure new instances using the Policy Server Configuration wizard. Running the wizard to configure new Web server instances can cause the existing configured Web server instance to fail.

How to use the Configuration Wizard

Complete the following procedures to use the Policy Server Configuration wizard:

1. Review the Configuration Wizard System Requirements.
2. Review the Policy Store Considerations.
3. Gather Information for the wizard if you are configuring:
   - The SOA Security Manager key database (smkeydatabase)
   - The OneView Monitor
   - An ADAM server policy store
   - A Sun Java Systems Directory Server policy store
4. Run the Configuration Wizard.

   Note: You can run the Configuration wizard in either GUI or console mode.

Configuration Wizard Requirements

Ensure you meet the following requirements before using the Policy Server Configuration wizard:

- The Policy Server Configuration wizard requires at least 150 MB of free space in /tmp.
- Run the wizard as a UNIX user with local administrator privileges.

  Run the wizard as a UNIX user that has sufficient privileges to modify the Web server's configuration files.

Policy Store Considerations

Consider the following before running the Policy Server installer or the Policy Server Configuration wizard:

- The Policy Server installer and configuration wizard can automatically configure the policy store in an Active Directory Application Mode (ADAM) or a Sun Java System Directory Server.

  Note: Ensure you have met the prerequisites for configuring ADAM as a policy store before running the installation.

- You may use any other supported LDAP or relational database as a policy store, but must manually configure the policy store after installing the Policy Server. Manually configuring an LDAP directory server or relational database as a policy store is detailed in this guide.

Note: The Policy Server installer and configuration wizard cannot automatically configure a policy store that is being connected to using an SSL connection.

More information:

ADAM Server Prerequisites (see page 121)

Gather Information for the Configuration Wizard

The Policy Server Configuration Wizard requires specific information to configure Policy Server components.

Note: Installation worksheets are provided to help you gather and record information prior to installing or configuring Policy Server components using the Policy Server Installation Wizard or the Policy Server Configuration Wizard. You may want to print these worksheets and use them to record required information prior to running either wizard.

Required Information

Gather the following required information before running the Policy Server installer or the Configuration wizard. You can use the Required Information Worksheet to record your values.

- **JRE location**—Identify the folder in which the installer can locate the supported JRE and ensure that the JRE is set correctly in the PATH variable. The installer cannot locate the JRE if the JRE is not set correctly in the PATH variable.
- **Policy Server Installation location**—Determine where the installer should install the Policy Server.

- **Encryption key value**—Determine the encryption key value. An *encryption key* is a case-sensitive, alphanumeric key that secures data sent between the Policy Server and the policy store. All Policy Servers that share a policy store must be configured using the same encryption key. For stronger protection, define a long encryption key.

  Limits: 6 to 24 characters.

More information:

Required Information Worksheet (see page 323)

SiteMinder Key Database Information

You only have to gather SOA Security Manager key database (smkeydatabase) information if you:

- Plan on using features related to eTrust SiteMinder FSS.
- Plan on configuring a SOA Security Manager Information Card Authentication scheme, for example, for the support of Microsoft CardSpace.

The Policy Server installer requires that you enter a password when configuring the smkeydatabase. The smkeydatabase password is used to encrypt the key and certificate data in the key database. You can use the SOA Security Manager Key Database Information Worksheet to record your value.

OneView Monitor Information

You only have to gather OneView Monitor information if you plan on configuring the OneView Monitor.

Gather the following required information to configure the OneView Monitor. You can use the OneView Monitor Information Worksheet to record your values.

- **JDK path**—Identify the path to the required JDK version.
- **ServletExec installation directory**—Identify ServletExec installation directory.

  Example: /usr/local/NewAtlanta/ServletExecAS
- **ServletExec port number**—Determine the port number for the ServletExec instance.
- **Sun Java System administrator directory**—Determine the following information:
  - The installed location of the Sun Java System.
  - The installed location of the Sun Java System Web servers.

  Example: */sunjavasystem_home/location*

sunjavasystem home

Specifies the installed location of the Sun Java System.

location

Specifies the installed location of the Sun Java System Web servers.

- **Multiple ServletExec instances**—If you have multiple ServletExec instances, determine the instance to which you want to configure the OneView Monitor GUI.

More information:

OneView Monitor Information Worksheet (see page 323)

ADAM Server Information

You only have to gather ADAM server information if you plan on using an ADAM server as a policy store.

Gather the following required information to configure an ADAM server as a policy store. You can use the ADAM Server Information Worksheet to record your values.

- **System IP address**—Identify the IP address of the system where the ADAM server is installed.
- **Server instance port number** - Determine the port number for the ADAM server instance.
- **Root DN of the application partition**—Identify the existing root DN location of the application partition in the ADAM server where the policy store schema data should be installed.

  Example: dc=ca,dc=com
- **ADAM administrator domain name**—Identify the full domain name, including the guid value, of the ADAM administrator.

  Example: CN=user1,CN=people,CN=Configuration,CN=<guid>
- **ADAM administrator password**—Identify the password for the ADAM administrator.

- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Identify the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.

- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

ADAM Server Information Worksheet (see page 324)

Sun Java System Directory Server Information

You only have to gather Sun Java System Directory Server information if you plan on configuring a Sun Java System Directory Server as a policy store.

Gather the following required information to configure a Sun Java System Directory Server as a policy store. You can use the Sun Java System Directory Server Information Worksheet to record your values.

- **System IP address**—Determine the IP address of the system where the Sun Java Systems Directory Server is installed.
- **Directory instance port number**—Determine the port number for the Sun Java Systems Directory Server instance.

  Default: 389
- **Root DN—**Identify the root DN of the Sun Java System Directory Server.

  Example: o=yourorg.com
- **Administrator account**—Identify the user name (Bind DN) for the LDAP administrator account.

  Example: cn=Directory Manager
- **Administrator password**—Identify the password for the Sun Java System Directory Server administrator.

- **Alternate LDAP administrator**—By default, SOA Security Manager uses the LDAP administrator account to communicate with the LDAP server. You can, however, have the SOA Security Manager policy store administered through a different LDAP user account. Determine the complete administrator DN and password to configure SOA Security Manager in this way.

  Note: This user should have all of the necessary privileges to modify attributes and change passwords.

- **SOA Security Manager Super User password**—The predefined SOA Security Manager Super User account has maximum SOA Security Manager privileges. Determine the password for the SOA Security Manager Super User account.

  Limits: 6 to 24 alphanumeric characters.

  Note: We recommend that this account not be used for day-to-day operations. Rather, use the Super User account to access the FSS Administrative UI for the first time and create an administrator account with system configuration privileges.

More information:

Sun Java System Directory Server Information Worksheet (see page 324)

Run the Configuration Wizard in GUI Mode

You run the Policy Server Configuration wizard to configure individual Policy Server components.

To run the Configuration wizard in GUI mode

1. Exit all running applications.
2. Execute the following script in a ksh shell from the SOA Security Manager installation directory:

   ```
   ../ca_ps_env.ksh
   ```

   Note: Be sure that there is a space between the periods (. .) when running the script.

3. Open a shell and run the following command:

   ```
   ./ca-ps-config.bin gui
   ```

   The Configuration wizard starts.

4. Use the system and component information you have gathered to configure a policy store and individual components.

   Note: When prompted to initialize the LDAP instance do so only to configure a new policy store instance.

5. Review the installation settings and click Install.

   The wizard configures the selected components to work with the Policy Server.

   Note: The installation can take several minutes.

6. Click Done.

   The components you selected are configured.

Note: If you experience problems, you can locate the installation log file and the policy store details file in *siteminder_home*/siteminder/install_config_info.

siteminder_home

Specifies the Policy Server installation path.

Run the Configuration Wizard in Console Mode

You run the Policy Server Configuration wizard to configure individual Policy Server components.

To run the Configuration wizard in console mode

1. Exit all running applications.
2. Execute the following script in a ksh shell from the SOA Security Manager installation directory:

   ```
   ../ca_ps_env.ksh
   ```

   Note: Be sure that there is a space between the periods (. .) when running the script.

3. Open a shell and run the following command:

   ```
   ./ca-ps-config.bin -i console
   ```

   The Configuration wizard starts.

4. Use the system and component information you have gathered to configure a policy store and individual components.

   Note: When prompted to initialize the LDAP instance do so only to configure a new policy store instance.

5. Review the installation settings and click Enter.

   The wizard configures the selected components to work with the Policy Server.

   Note: The installation can take several minutes.

6. Press Enter

   The installer closes. The selected components are configured.

Note: If you experience problems, you can locate the installation log file and the policy store details file in *siteminder_home*/siteminder/install_config_info.

siteminder_home

Specifies the Policy Server installation path.

Backup Versions of Obj.conf and Magnus.conf Files

Each time you run the Policy Server Configuration Wizard, it creates backup versions of the obj.conf and magnus.conf files. These files allow you to return to the original Web server configuration you had before running the Policy Server Configuration Wizard or installation program. These backup versions are in the following format in the Web server's configuration directory:

obj.conf.<*year*>-<*month*>-<*date*>-<*hour*>-<*minutes*>-<*seconds*>.bak

magnus.conf.<*year*>-<*month*>-<*date*>-<*hour*>-<*minutes*>-<*seconds*>.bak

Example backup version are obj.conf.2003-11-25-16-58-47.bak and magnus.conf.2003-11-25-17-07-11.bak.

How to Uninstall the Policy Server

Complete the following procedures to uninstall the Policy Server:

1. Shut down all instances of the Policy Server Management Console.

   Note: More information on shutting down the Policy Server Management Console exists in the *Policy Administration Guide*.

2. Set the JRE in the Path Variable.
3. Remove Policy Server References from Agent Host Files.
4. Stop all SOA Security Manager Processes.
5. Uninstall the Policy Server.
6. Remove SOA Security Manager References from IWS.
7. Remove SOA Security Manager References from ServletExec/AS

Remove Policy Server References from Agent Host Files

You remove the Policy Server reference from the SmHost.conf file to prevent unexpected results from the Web Agent once the Policy Server is uninstalled.

To remove the Policy Server reference

1. Navigate to *web_agent_home*/config.

   web_agent_home

   Specifies the installation directory of the Web Agent.

2. Open the SmHost.conf file in a text editor.
3. Delete the line that begins with "policyserver=".

   Note: This line contains the IP address and port numbers for the Policy Server you are uninstalling.

4. Save SmHost.conf.

   The SmHost.conf file no longer references the Policy Server you are uninstalling.

Set the JRE in the PATH Variable

You set the JRE in the PATH variable when uninstalling the Policy Server, Web Agent, SDK, or documentation to prevent the uninstallation program from stopping and issuing error messages.

To set the JRE in the PATH variable

1. Run the following command:

   PATH=$PATH:<JRE>/bin

   JRE

   Specifies the location of the JRE.

2. Run the following command:

   export PATH

   The JRE is set in the PATH variable.

Stop all SOA Security Manager Processes

You stop all SOA Security Manager processes to ensure that Policy Server files are safely removed.

To stop all SOA Security Manager processes

1. Log into the UNIX system with the smuser account.
2. Run stop-all, which is located in the /siteminder directory.

   All SOA Security Manager processes stop.

Uninstall the Policy Server

To uninstall the Policy Server when it is no longer required on the system, run the SOA Security Manager uninstall wizard.

To uninstall SOA Security Manager components on Windows or UNIX systems

1. Navigate to the *SOA_HOME*\install_config_info (Windows) or *SOA_HOME*/install_config_info (UNIX) directory and run the SOA Security Manager uninstall wizard to remove core SOA Security Manager components:
   - Windows: soa-uninstall.cmd
   - UNIX: soa-uninstall.sh

   SOA_HOME

   Specifies the SOA Security Manager installation location.

   The uninstall wizard starts.
2. Choose whether you want to perform a complete uninstall or whether to uninstall specific features and proceed.
3. If you chose to uninstall only specific features, select the installed components that you want to uninstall and proceed.

The uninstall wizard removes all selected SOA Security Manager components.

1.

2.

3.

4.

Remove SOA Security Manager References from IWS

You manually remove SOA Security Manager references from IWS after uninstalling the Policy Server. SOA Security Manager references are left in the obj.conf file and the magnus.conf file.

To remove SOA Security Manager references from IWS

1. Log into an account that has privileges to access and modify the Web server's configuration.
2. Go to the following at the Solaris command line.

   <*SunJavaSystem_home*>/https-<*hostname*>/config

   The obj.conf and magnus.conf files appear in the config folder.
3. Open obj.conf and remove the following lines:

   NameTrans fn="assign-name" from="/servlet/*" name="<*ServletExec_instance name*>"
   NameTrans fn="assign-name" from="*.jsp*" name="<ServletExec_instance name>"
   NameTrans fn="pfx2dir" from="/sitemindermonitor" dir="/<*siteminder_installation*>/monitor"
   NameTrans fn="pfx2dir" from="/siteminderdcgi" dir="/<*siteminder_installation*>/admin" name="cgi"
   NameTrans fn="pfx2dir" from="/siteminder" dir="/<*siteminder_installation*>/admin"
   NameTrans fn="pfx2dir" from="/netegrity_docs" dir="/netegrity/netegrity_documents"
   <Object name="<ServletExec_instance name>">
   Service fn="ServletExecService" group="<ServletExec_instance name>"
   </Object>
4. Save and close the obj.conf file.
5. Open magnus.conf and remove the following lines:

   Init fn="init-cgi" SM_ADM_UDP_PORT="44444" SM_ADM_TCP_PORT="44444"
   Init fn="load-modules" shlib="/<Servlet_Exec_Install>/bin/ServletExec_Adapter.so"
   funcs="ServletExecInit,ServletExecService"
   Init fn="ServletExecInit" <ServletExec_instance name>.instances="<IP_Address>:<port_number>"
6. Save and close magnus.conf.
7. Restart the Web server.

   SOA Security Manager references are removed from IWS.

   The SOA Security Manager references no longer appear in IWS.

Remove SOA Security Manager References from StartServletExec

You manually remove SOA Security Manager references from StartServletExec after uninstalling the Policy Server.

To remove SOA Security Manager references from StartServletExec

1. Log into an account that has privileges to access and modify the ServletExec configuration.

   Go to the following at the Solaris command line:
   /usr/NewAtlanta/ServletExecAS/<*ServletExec_instance name*> folder

2. Remove the following lines from the StartServletExec script:

   CLASSPATH=${NA_ROOT}/lib/ServletExec42.jar:${NA_ROOT}/lib/servlet.jar:${JL}/tools.jar:${NA_ROOT}/lib/jaxp.jar:${NA_ROOT}/lib/crimson.jar:${NA_ROOT}/lib/jndi.jar:${NA_ROOT}/se-${SEINSTANCE}/classes
   CLASSPATH=${NA_ROOT}/lib/ServletExec42.jar:${NA_ROOT}/lib/servlet.jar:${JL}/tools.jar:${NA_ROOT}/lib/jaxp.jar:${NA_ROOT}/lib/crimson.jar:${NA_ROOT}/lib/jndi.jar:${NA_ROOT}/se-*${SEINSTANCE}/classes:/export/smuser/siteminder/monitor/smmongui.jar:/<siteminder_installation>*/lib/smconapi.jar:/e<*siteminder_installation*>/lib/smmonclientapi.jar
   $SENAME $HOMEDIR $MIMEFILE $DOCROOTDIR -allow 127.0.0.1 -port $PORT $SEOPTS"
   $SENAME $HOMEDIR $MIMEFILE $DOCROOTDIR -allow 127.0.0.1 -port $PORT $SEOPTS -addl "/siteminder*monitor=/<siteminder_installation>*/monitor""

3. Save and close the StartServletExec script.
4. Restart ServletExec.

   SOA Security Manager references no longer exist in the StartServletExec.

You have finished uninstalling the Policy Server.

Scripting Interface

The Command Line Interface allows you to write Perl scripts to configure and manage policy stores. The installation program installs a full version of Perl and puts the interface files in the *siteminder_installation*/CLI directory.

siteminder_installation

Specifies the installed location of SOA Security Manager.

Example: /home/smuser/siteminder/CLI

To use the Command Line Interface, make sure the following directory is in your system's PATH environment variable before any other Perl bin directories on your machine.

For example: /home/smuser/siteminder/CLI/bin

Note: More information on the scripting interface exists in the *Programming Guide for Perl*

Chapter 6: Configuring LDAP Directory Servers as a Policy or Key Store

This section contains the following topics:

LDAP Directory Servers as a Policy or Key Store

The SOA Security Manager policy store is the repository for all policy related information. All Policy Servers in a SOA Security Manager installation must share the same policy store data, either directly or via replication. SOA Security Manager is installed with tools that let administrators move policy store data from one storage facility to another.

When you install the Policy Server, the default policy store is a Sun Java System (formerly Sun ONE/iPlanet) Directory Server or Microsoft Active Directory Application Mode. You can configure the Policy Server to use another LDAP directory, a SQL Server database, or an Oracle database as a policy store after you have completed the Policy Server installation. Also, after installation, you can use the Policy Server Management Console's Data tab to have the Policy Server point to another policy store.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Installation Road Map

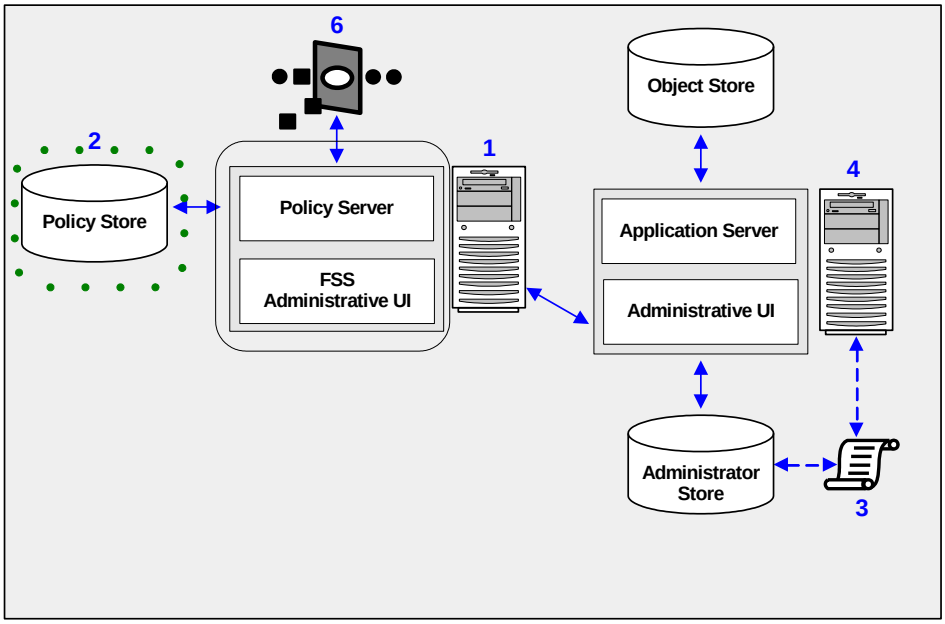
The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- The components surrounded by a solid line represent the Policy Server and a non-registered FSS Administrative UI. A Policy Server must be installed before configuring a policy store. Do not continue with the installation process if a Policy Server is not yet part of your environment.

  Note: The FSS Administrative UI is installed with the Policy Server. Unless you need to generate WS-Security SAML assertion tokens, you may safely leave the eTrust SiteMinder FSS on the Policy Server machine without registering it with the Policy Server.

- The component surrounded by a green dotted line represents the policy store, which you configure at this point in the installation process. The policy store contains all of the Policy Server data. You configure a policy store in either an LDAP or relational database.

  Note: The following figure depicts a single policy/key store instance. Although not illustrated, your environment may use separate instances for individual policy and key stores.



The following sections in the documentation detail how to:

- Configure the policy store.
- Establish a connection to the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Important Considerations

To avoid possible policy store corruption, ensure that the server on which the policy store will reside is configured to store objects in UTF-8 form. For more information on configuring your server to store objects in UTF-8 form, see the documentation for that server.

CA Directory as a Policy Store

Policy Servers installed on either Windows or UNIX systems can use CA Directory as a policy store. The following sections detail how to configure your directory server as a policy store.

Gather Directory Server Information

Configuring a CA Directory as a policy store requires specific directory server information.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

Gather the following information before configuring the policy store. You can use the Policy Store Worksheets to record your values.

- **Host information—**Determine the fully qualified host name or the IP address of the system on which CA Directory is running.
- **DSA port number**—Determine the port on which the DSA is to listen.
- **Base DN**—Determine the distinguished name of the node in the LDAP tree in which policy store objects are to be defined.

- **Administrative DN**—Determine the LDAP user name of the user with the privileges to create, read, modify, and delete objects in the DSA.
- **Administrative password**—Determine the password for the administrative user.

More information:

CA Directory Information Worksheet (see page 325)

How to Configure the Policy Store

To configure CA Directory as policy store, complete the following procedures:

1. Create a DSA for the Policy Store.
2. Create the Policy Store Schema.
3. Open the DSA.
4. Create the Base Tree Structure for Policy Store Data.
5. Create a Super User Administrator for the DSA.
6. Point the Policy Server at the Policy Store.
7. Set the SOA Security Manager Super User Password.
8. Verify the CA Directory Cache Configuration.
9. Import the Default Policy Store Objects.
10. Import the Policy Store Data Definitions.

Create a DSA for the Policy Store

Create the DSA by running one of the following commands:

- **r8.1**

  dxnewdsa *DSA_Name Database_Name port* c *country_code* o *DSA_Name*

  Note: The command creates the database if it does not exist.

  DSA_Name

  Specifies the name of the DSA.

  Database_Name

  Specifies the name of the new or existing database.

  Note: The command creates the database if it does not exist.

port

Specifies the port on which the DSA is to listen.

c *country_code* o *DSA_Name*

Specifies the DSA prefix.

Example: c US o psdsa

- **r12.x**

dxnewdsa *DSA_Name port* "o=*DSA_Name*,c=*country_code*"

DSA_Name

Specifies the name of the DSA.

port

Specifies the port on which the DSA is to listen.

o=*DSA_Name*,c=*country_code*

Specifies the DSA prefix.

Example: "o=psdsa,c=US"

The dxnewdsa utility starts the new DSA.

Note: If the DSA does not automatically start, run the following:

dxserver start *DSA_Name*

Create the Policy Store Schema

You create the policy store schema so the directory server can function as a policy store and store SOA Security Manager objects.

To create the Policy Store schema

1. Copy the following files into the CA Directory *DXHOME*\config\schema directory:
   - netegrity.dxc
   - etrust.dxc

   DXHOME

   Specifies the Directory Server installation path.

   Note: The netegrity.dxc file is installed with the Policy Server in *policy_server_home*\eTrust. The etrust.dxc file is installed with the Policy Server in *siteminder_home*\xps\db.

   siteminder_home

   Specifies the policy server installation path.

2. Copy the following files into the CA Directory *DXHOME*\bin directory.
   - etrust_schema.txt
   - schema.txt

   Note: The etrust_schema.txt file is installed with the Policy Server in *policy_server_home*\xps\db. The schema.txt file is installed with the Policy Server in *siteminder_home*\eTrust.
3. Create a new SOA Security Manager schema file by copying the default.dxg schema file and renaming it.

   Note: The default.dxg schema file is located in *DXHOME*\config\schema\default.dxg.

   Example: copy the default.dxg schema file and rename the copy to smdsa.dxg
4. Add the following lines to the bottom of the new SOA Security Manager schema file:

   ```
   #CA Schema

   source "netegrity.dxc";

   source "etrust.dxc";
   ```
5. Edit the DSA's DXI file (*DSA_Name*.dxi) by changing the schema from default.dxg to the new SOA Security Manager schema file.

   DSA_Name

   Represents the name of the DSA you created using the dxnewdsa utility.

   Note: the DSA's DXI file is located in *DXHOME*\config\servers.
6. Add the following lines to the end of the DSA's DXI file:
   - **r8.1**

     ```
     # cache configuration
     set max-cache-size = 100;
     set cache-index = all-attributes;
     set cache-attrs = all-attributes;
     set cache-load-all = true;
     set lookup-cache = true;
     set ignore-name-bindings = true;
     ```

- **r12.x**

```
# cache configuration
set max-cache-size = 100;
set cache-attrs = all-attributes;
set cache-load-all = true;
set ignore-name-bindings = true;
```

Note: The max-cache-size entry is the total cache size in MB. Adjust this value based on the total memory available on the CA Directory server and overall size of the policy store.

7. Open the DSA's default DXC file (default.dxc).

   Note: The default DXC file is located in *DXHOME*\dxserver\config\limits.

8. Edit the settings to match the following:

   - **r8.1**

```
# size limits
set max-users = 1000;
set credits = 5;
set max-local-ops = 1000;
set max-dsp-ops = 1000;
set max-op-size = 1000;
set multi-write-queue = 20000;
```

   - **r12.x**

```
# size limits
set max-users = 1000;
set credits = 5;
set max-local-ops = 1000;
set max-op-size = 1000;
set multi-write-queue = 20000;
```

   Note: Editing the size limits settings prevents cache size errors from appearing in your CA Directory log files.

9. Save the DXC file.

10. As the DSA user, stop and restart the DSA using the following commands:

```
dxserver stop DSA_Name
dxserver start DSA_Name
```

DSA_Name

Specifies the name of the DSA.

The policy store schema is created.

Open the DSA

You create a view into the directory server to manage objects.

To create a view into the CA Directory Server

1. Launch the JXplorer GUI.
2. Select the connect icon.

   Connection settings appear.
3. Enter *host_name_or_IP_address* in the Host Name field.

   host_name_or_IP_address

   Specifies the host name or IP address of the system where CA Directory is running.
4. Enter *port_number* in the Port number field.

   port_number

   Specifies the port on which the DSA is listening.
5. Enter o=*DSA_Name*,c=*country_code* in the Base DN field.

   Example: o=psdsa,c=US
6. Select Anonymous from the Level list, and click Connect

   A view into DSA appears.

Create the Base Tree Structure for Policy Store Data

You create a base tree structure to hold policy store data. You use the JXplorer GUI to create the organizational units.

To create the base tree structure for policy store data

1. Select the root element of your DSA.
2. Create an organizational unit under the root element called:

   Netegrity
3. Create an organizational unit (root element) under Netegrity called:

   SiteMinder
4. Create an organizational unit (root element) under SiteMinder called:

   PolicySvr4

   The base tree structure is created.

Create a Super User Administrator for the DSA

You only have to create a Super User administrator if you do not have an administrator name and password that has the rights to create, delete, or modify the dsa. The Policy Server requires this information to connect to the policy store.

Use JXplorer to create an administrator that has the rights to create, delete, and modify objects in the DSA.

Note: Take note of administrator user name and password. You will use this information when pointing the Policy Server to the policy store.

Example: dn: cn=admin,o=yourcompany,c=in

Point the Policy Server to the Policy Store

You point the Policy Server to the policy store so the Policy Server can access the policy store.

To point the Policy Server to the policy store

1. Open the Policy Server Management Console.
2. Click the Data tab.

   Database settings appear.
3. Select Policy Store from the Database list.
4. Select LDAP from the Storage list.
5. Configure the following settings in the LDAP Policy Store group box.
   - LDAP IP Address
   - Admin Username
   - Password
   - Confirm Password

   Note: You can click Help for a description of fields, controls, and their respective requirements.
6. Click Apply.

   The policy store settings are saved.
7. Click Test LDAP Connection.

   SOA Security Manager returns a confirmation that the Policy Server can access the policy store.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Verify the CA Directory Cache Configuration

You can verify that the DXcache settings are enabled using the DXconsole.

Note: By default, the DxConsole is only accessible from localhost. For more information on using the set dsa command to let the DxConsole accept a connection from a remote system, refer to the *eTrust Directory Reference Guide*.

To verify that the cache is enabled

1. From a command prompt, enter the following to telnet to the DSA DXConsole port:

   telnet *DSA_Host DXconsole_Port*

   DSA_Host

   Specifies the host name or IP address of the system hosting the DSA.

   Note: If you are on the localhost, enter **localhost**. Entering a host name or IP Address results in a failed connection.

   DXConsole_Port

   Specifies the port on which the DXconsole is listening.

   Default: The DXconsole port is set to the value of the DSA port +1.

   Example: If the DSA is running on port 19389, the DXconsole port is 19390.

   The DSA Management Console appears.

2. Enter the following command:

   get cache;

   The DSA Management Console displays the current DSA DXcache settings and specifies if directory caching is enabled.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

   -v

   Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

   Default value: stdout

   -cf

   (Optional) Imports sensitive data using FIPS-compatible cryptography.

   Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

   smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.

3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.

4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.

5. Run the following command:

   XPSDDInstall FssSmObjects.xdd

6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Sun Java System Directory Server as a Policy Store

Policy Servers installed on either Windows or UNIX systems can use a Sun Java System Directory Server as a policy store. The following sections detail how to manually configure your directory server as a policy store.

Note: You can use the Policy Server Configuration wizard to configure this type of LDAP directory server as a policy store automatically.

Gather Directory Server Information

Configuring an LDAP directory server as a policy store or upgrading an existing policy store requires specific directory server information. Gather the following information before beginning. You can use the Policy Store Worksheets to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

Host information

Specifies the fully-qualified host name or the IP Address of the directory server.

Port information

(Optional) Specifies a non-standard port.

Default values: 636 (SSL) and 389 (non-SSL)

Administrative DN

Specifies the LDAP user name of a user who has privileges to create, read, modify, and delete objects in the LDAP tree underneath the policy store root object.

Administrative password

Specifies the password for the Administrative DN.

Policy store root DN

Specifies the distinguished name of the node in the LDAP tree where policy store objects are to be defined.

SSL client certificate

Specifies the pathname of the directory where the SSL client certificate database file resides.

Limit: SSL only

More information:

Active Directory Information Worksheet (see page 326)
Policy and Data Store Worksheets (see page 325)
Sun Java System Directory Server Information Worksheet (see page 326)

How to Configure the Policy Store

To manually configure a Sun Java Systems Directory Server as a policy store, complete the following procedures:

1. (Optional) If applicable, use the LDAP vendor software to create an LDAP Directory Server instance.
2. (Optional) If applicable, review the Sun One 5.x Directory Server Considerations.
3. (Optional) If applicable, use the LDAP vendor software to create a user with privileges to create, read, modify, and delete objects in the LDAP tree underneath the policy store root object.
4. Point the Policy Server to the Directory Server.

5. Create the Policy Store Schema.
6. Set the SOA Security Manager Super User Password.
7. Import the Default SOA Security Manager Objects.
8. Import the Policy Store Data Definitions.
9. Restart the Policy Server.

Sun ONE LDAP Directory Server Considerations

Consider the following if you plan on using a Sun ONE 5.x directory server as a policy store.

smldapsetup and Sun Java System Directory Server Enterprise Edition

In a Sun Java System Directory Server Enterprise Edition (formerly Sun ONE/iPlanet) directory server, smldapsetup creates the ou=Netegrity, *root* sub suffix and PolicySvr4 database.

root

The directory root you specified in the Root DN field on the Data tab of the Policy Server Management Console. This variable has to be either an existing root suffix or sub suffix.

Example: If your root suffix is dc=netegrity,dc=com then running smldapsetup produces the following in the directory server:

- A root suffix, dc=netegrity,dc=com, with the corresponding userRoot database.
- A sub suffix, ou=Netegrity,dc=netegrity,dc=com, with the corresponding PolicySvr4 database.

Example: If you want to place the policy store under ou=apps,dc=netegrity,dc=com, then ou=apps,dc=netegrity,dc=com has to be either a root or sub suffix of the root suffix dc=netegrity,dc=com.

If it is a sub suffix, then running smldapsetup produces the following:

- A root suffix, dc=netegrity,dc=com, with the corresponding userRoot database.
- A sub suffix, ou=apps,dc=netegrity,dc=com, with the corresponding Apps database.
- A sub suffix, ou=Netegrity,ou=apps,dc=netegrity,dc=com, with the corresponding PolicySvr4 database.

Note: For more information about root and sub suffixes, see the Sun Microsystems documentation.

Replicate the Policy Store on Sun Java System Directory Server Enterprise Edition

For Sun Java System Directory Server Enterprise Edition (formerly Sun ONE/iPlanet), SOA Security Manager r12.1 creates a UserRoot and a PolicySvr4 database. The PolicySvr4 database has suffix mappings pointing to it. To replicate this policy store, set up a replication agreement for the PolicySvr4 database directory.

Note: More information about a replication agreement, see the Sun Microsystems documentation.

After you create the replication agreement, replicate the SOA Security Manager indexes.

To replicate SOA Security Manager indexes

1. Generate the SOA Security Manager indexes:

   smldapsetup ldgen -x -findexes.ldif

   Important! If you are running a SOA Security Manager utility or executable on Windows Server 2008, be sure to open the command–line window with Administrator permissions, even if you are logged into the system as an Administrator. For more information, see the release notes for your SOA Security Manager component.

2. Set up the indexes on a replica server:

   smldapsetup ldmod -x -findexes.ldif -h*host* -p*replicaport* -d*AdminDN* -w*AdminPW*

   host

   Specifies the replica host.

   replicaport

   Specifies the replica port number.

   AdminDN

   Specifies the replica administrator DN.

   Example: cn=directory manager

   AdminPW

   Specifies the replica administrator password.

   The SOA Security Manager indexes are replicated.

Point the Policy Server to the Directory Server

You point the Policy Server to the LDAP directory server so that the Policy Server has the necessary system information and administrative privileges to read and write information to the policy store.

To point the Policy Server to the directory server

1. Run the following command from the Policy Server host system:

   smldapsetup status -h *host* -p *port* -d *AdminDN*
   -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

   -h *host*

   Specifies the IP Address of the LDAP server.

   -p *port*

   Specifies the port number of the LDAP server.

   -d *AdminDN*

   Specifies the name of a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   ADAM: Specifies the full domain name, including the guid value, of the ADAM administrator.

   Example: CN=user1,CN=People,CN=Configuration,CN,{guid}

   -w *AdminPW*

   Specifies the password for a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   -r *root*

   Specifies the DN location of the SOA Security Manager data in the LDAP directory.

   ADAM: Specifies the existing root DN location of the application partition in the ADAM server where you want to put the policy store schema data.

   -ssl 1|0

   Specifies an SSL connection.

   Limits: 0=no | 1=yes

   Default: 0

-c *cert*

(Only required if the ssl value is 1) Specifies the path to the directory where the SSL client certificate database file, cert7.db, exists.

The correct configuration of the LDAP policy store connection parameters is verified.

2. Run the following command:

smldapsetup reg -h *host* -p *port* -d *AdminDN* -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

The connection to the LDAP directory server is tested and the server is configured as a SOA Security Manager policy store.

Create the Policy Store Schema

You create the policy store schema so the directory server can function as a policy store and store SOA Security Manager objects.

To create the policy store schema

1. Run the following command from the Policy Server host system:

smldapsetup ldgen -f*file_name*

file_name

Specifies the name of the LDIF file you are creating.

An LDIF file with the SOA Security Manager schema is created.

2. Run the following command:

smldapsetup ldmod -f*file_name*

file_name

Specifies the name of the LDIF you created.

smldapsetup imports the policy store schema.

3. Run the following command:

smldapsetup ldmod -f*policy_server_home*/xps/db/SunOne.ldif

policy_server_home

Specifies the Policy Server Installation path.

The policy store schema is extended. You have created the policy store schema.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

   -v

   Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

   Default value: stdout

   -cf

   (Optional) Imports sensitive data using FIPS-compatible cryptography.

   Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

   smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.

3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.

4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.

5. Run the following command:

   XPSDDInstall FssSmObjects.xdd

6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Active Directory as a Policy Store

Microsoft Active Directory is the native LDAP-compatible directory for Windows. Policy Servers installed on either Windows or UNIX systems can use Active Directory as a policy store. Moreover, the Policy Server and policy store can be installed on separate machines. For example, a Policy Server installed on a UNIX machine can use an Active Directory policy store on a Windows system.

Note: If Active Directory is to communicate with the Policy Server over SSL, ensure that the SSL client certificate contains the CN of the SubjectDN. The Policy Server crashes if the SSL client certificate does not contain this information.

The following sections detail how to configure your directory server as a policy store.

Gather Directory Server Information

Configuring an LDAP directory server as a policy store or upgrading an existing policy store requires specific directory server information. Gather the following information before beginning. You can use the Policy Store Worksheets to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

Host information

Specifies the fully-qualified host name or the IP Address of the directory server.

Port information

(Optional) Specifies a non-standard port.

Default values: 636 (SSL) and 389 (non-SSL)

Administrative DN

Specifies the LDAP user name of a user who has privileges to create, read, modify, and delete objects in the LDAP tree underneath the policy store root object.

Administrative password

Specifies the password for the Administrative DN.

Policy store root DN

Specifies the distinguished name of the node in the LDAP tree where policy store objects are to be defined.

SSL client certificate

Specifies the pathname of the directory where the SSL client certificate database file resides.

Limit: SSL only

More information:

Active Directory Information Worksheet (see page 326)
Policy and Data Store Worksheets (see page 325)
Sun Java System Directory Server Information Worksheet (see page 326)

How to Configure the Policy Store

To manually configure an Active Directory directory server as a policy store, complete the following procedures:

1. (Optional) If applicable, use the LDAP vendor software to create an LDAP Directory Server instance.
2. (Optional) If applicable, use the LDAP vendor software to create a user with privileges to create, read, modify, and delete objects in the LDAP tree underneath the policy store root object.

   Note: You do not have to complete this procedure if you have gathered the LDAP user name of a user with privileges to create, read, modify, and deleting objects in the LDAP tree underneath the policy store root object.
3. Point the Policy Server to the Directory Server.
4. Create the Policy Store Schema.
5. Set the SiteMinder Super User Password.
6. Import the Default SOA Security Manager Objects.
7. Import the Policy Store Data Definitions.
8. Restart the Policy Server.

Point the Policy Server to the Directory Server

You point the Policy Server to the LDAP directory server so that the Policy Server has the necessary system information and administrative privileges to read and write information to the policy store.

To point the Policy Server to the directory server

1. Run the following command from the Policy Server host system:

   smldapsetup status -h *host* -p *port* -d *AdminDN*
   -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

   -h *host*

   Specifies the IP Address of the LDAP server.

   -p *port*

   Specifies the port number of the LDAP server.

   -d *AdminDN*

   Specifies the name of a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   ADAM: Specifies the full domain name, including the guid value, of the ADAM administrator.

   Example: CN=user1,CN=People,CN=Configuration,CN,{guid}

   -w *AdminPW*

   Specifies the password for a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   -r *root*

   Specifies the DN location of the SOA Security Manager data in the LDAP directory.

   ADAM: Specifies the existing root DN location of the application partition in the ADAM server where you want to put the policy store schema data.

   -ssl 1|0

   Specifies an SSL connection.

   Limits: 0=no | 1=yes

   Default: 0

-c *cert*

(Only required if the ssl value is 1) Specifies the path to the directory where the SSL client certificate database file, cert7.db, exists.

The correct configuration of the LDAP policy store connection parameters is verified.

2. Run the following command:

smldapsetup reg -h *host* -p *port* -d *AdminDN* -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

The connection to the LDAP directory server is tested and the server is configured as a SOA Security Manager policy store.

Create the Policy Store Schema

You create the policy store schema so the directory server can function as a policy store and store SOA Security Manager objects.

To create the policy store schema

1. Run the following command from the Policy Server host system:

smldapsetup ldgen -f*file_name*

file_name

Specifies the name of the LDIF file you are creating.

An LDIF file with the SOA Security Manager schema is created.

2. Run the following command:

smldapsetup ldmod -f*file_name*

file_name

Specifies the name of the LDIF you created.

smldapsetup imports the policy store schema.

3. Navigate to *policy_server_home*\xps\db and open the following file:

ActiveDirectory.ldif

4. Manually replace each instance of <RootDN> with the actual value of the root DN.

   Example: dc=domain,dc=com

5. Run the following command:

   smldapsetup ldmod -f*policy_server_home*/xps/db/ActiveDirectory.ldif

   policy_server_home

   Specifies the Policy Server installation path.

   The policy store schema is extended. You have created the policy store schema.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-cf

(Optional) Imports sensitive data using FIPS-compatible cryptography.

Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
-d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.
2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.
3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.
4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.
5. Run the following command:

   XPSDDInstall FssSmObjects.xdd
6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Support for Active Directory ObjectCategory Indexing Attribute

Unlike other LDAP-compatible directories, Active Directory does not index policy store objects using the objectClass attribute by default. Instead, the objects are indexed by the objectCategory attribute. To enhance searches, you can either configure objectClass as an indexable attribute (see the Active Directory documentation) or enable objectCategory support in the Policy Server.

Enable or Disable ObjectCategory Attribute Support

On Windows Systems:

To enable or disable ObjectCategory attribute support

1. Launch the Windows Registry Editor.
2. Locate the key HKLM\Software\Netegrity\SiteMinder\CurrentVersion\DS\LDAPProvider.
   a. To enable support, set the EnableObjectCategory value to 1.
   b. To disable support, set the EnableObjectCategory value to 0.

      Note: The default value is 0.

On UNIX systems:

To enable or disable ObjectCategory attribute support

1. In a text editor, open the SOA Security Manager sm.registry file, located in *<siteminder_installation>*/registry.
2. Locate the key

   HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider.

   a. To enable support, set the EnableObjectCategory value to 1.

   b. To disable support, set the EnableObjectCategory value to 0.

   Note: The default value is 0.

Microsoft ADAM as a Policy Store

Policy Servers installed on either Windows or UNIX systems can use Microsoft ADAM as a policy store. The following sections detail how to manually configure your directory server as a policy store.

Note: You can use the Policy Server Configuration wizard to configure this type of LDAP directory server as a policy store automatically.

ADAM Server Prerequisites

Before you configure an ADAM policy store, ensure you have met the following prerequisites:

1. Path the ADAM Server.
2. Create a Policy Store Partition for the Administrative User.

Patch the ADAM Server

Apply the Microsoft patch Q840991 to the ADAM server. This patch lets you create users in the configuration partition. Only users with administrative rights in this partition can import the policy store schema. You can download the patch at www.microsoft.com or by contacting Microsoft Product Support.

Create a Policy Store Partition for the Administrative User

You create a policy store partition and add an administrative user to it to ensure the policy store schema can be imported. Only an administrative user in the configuration partition can import the policy store schema. This user must have administrative rights over the configuration partition and all of the application partitions, including the policy store partition.

Note: The following procedure assumes you are familiar with configuration, application, and schema partitions. More information exists at:

http://www.c-sharpcorner.com/Code/2004/Aug/DirectoryServices.asp

To create a policy store partition for the administrative user

1. Click Start, Program Files, ADAM, ADAM ADSI Edit.

   The ADAM ADSI Edit utility opens.

2. Create a policy store partition.

3. Navigate to the following in the configuration partition:

   cn=directory service, cn=windows nt,

   cn=services, cn=configuration, cn={guid}

4. Locate the msDS-Other-Settings attribute.

5. Add a new value to the msDS-Other-Settings attribute:

   ADAMAllowADAMSecurityPrincipalsInConfigPartition=1

6. In the configuration and policy store application partitions:

   a. Navigate to CN=Administrators, CN=Roles.

   b. Open the properties of CN=Administrators.

   c. Edit the member attribute.

   d. Click Add ADAM Account, and paste the full DN of the user you created in the configuration partition.

   e. Go to the properties of the user created and check the value set for the object "msDS-UserAccountDisabled". Ensure that the value is set false.

   The administrative user has rights over the configuration partition and all of the application partitions, including the policy store partition.

Gather Directory Server Information

Configuring an ADAM directory server as a policy store requires specific directory server information.

Gather the following information before configuring the policy store. You can use the Policy Store Worksheets to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Host information** —Determine the fully qualified host name or the IP Address of the directory server.
- **Port information** —Determine if directory server is using a non-standard port. The tools used to configure a policy store uses port 389 (non-SSL) and 636 (SSL) if port information is not provided.
- **Administrative DN** —Determine the full domain name, including the guid value, of the ADAM administrator.

  Example: CN=user1,CN=People,CN=Configuration,CN,{guid}
- **Administrative Password** —Determine the password for the Administrative DN.
- **Policy Store Root DN** —Determine the existing root DN location of the application partition in the ADAM server where you want to put the policy store schema data under.
- (Optional) **SSL client certificate** —If the directory connection is made over SSL, determine the path of the directory where the SSL client certificate database file exists.

More information:

Policy and Data Store Worksheets (see page 325)
Microsoft ADAM Information Worksheet (see page 327)

How to Configure the Policy Store

To manually configure ADAM as a policy store, complete the following procedures:

1. Ensure you have met the ADAM Server Prerequisites.
2. Ensure that you have gathered the necessary information.
3. Point the Policy Server to the Directory Server.
4. Create the Policy Store Schema.
5. Set the SiteMinder Super User Password.
6. Import the Default SOA Security Manager Objects.
7. Import the Policy Store Data Definitions.
8. Restart the Policy Server.

Point the Policy Server to the Directory Server

You point the Policy Server to the LDAP directory server so that the Policy Server has the necessary system information and administrative privileges to read and write information to the policy store.

To point the Policy Server to the directory server

1. Run the following command from the Policy Server host system:

   smldapsetup status -h *host* -p *port* -d *AdminDN*
   -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

   -h *host*

   Specifies the IP Address of the LDAP server.

   -p *port*

   Specifies the port number of the LDAP server.

   -d *AdminDN*

   Specifies the name of a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   ADAM: Specifies the full domain name, including the guid value, of the ADAM administrator.

   Example: CN=user1,CN=People,CN=Configuration,CN,{guid}

   -w *AdminPW*

   Specifies the password for a LDAP user with privileges to create a new LDAP schema in the LDAP directory server.

   -r *root*

   Specifies the DN location of the SOA Security Manager data in the LDAP directory.

   ADAM: Specifies the existing root DN location of the application partition in the ADAM server where you want to put the policy store schema data.

   -ssl 1|0

   Specifies an SSL connection.

   Limits: 0=no | 1=yes

   Default: 0

-c *cert*

(Only required if the ssl value is 1) Specifies the path to the directory where the SSL client certificate database file, cert7.db, exists.

The correct configuration of the LDAP policy store connection parameters is verified.

2. Run the following command:

smldapsetup reg -h *host* -p *port* -d *AdminDN* -w *AdminPW* -r *root* -ssl *1/0* -c *cert*

The connection to the LDAP directory server is tested and the server is configured as a SOA Security Manager policy store.

Create the Policy Store Schema

You create the policy store schema so the directory server can function as a policy store and store SOA Security Manager objects.

To create the policy store schema

1. Navigate to *policy_server_home*/bin or *policy_server_home*\bin from a command window.

policy server home

Specifies the Policy Server installation path.

2. Run the following command:

smldapsetup ldgen -f*file_name*

file name

Specifies the name of the LDIF file you are creating.

An LDIF file with the SOA Security Manager schema is created.

3. Run the following command:

smldapsetup ldmod -f*file_name*

file name

Specifies the name of the LDIF you created.

smldapsetup imports the policy store schema.

4. Navigate to *policy_server_home*/xps/db or *policy_server_home*\xps/db, and open the ADAM.ldif file.

5. Replace each instance of {guid} with the actual value of guid in braces, and save the file.

Example: {CF151EA3-53A0-44A4-B4AC-DA0EBB1FF200}

6. Navigate to *policy_server_home*/bin or *policy_server_home*\bin from a command window.
7. Run the following command:

   smldapsetup ldmod -f*policy_server_home*/xps/db/ADAM.ldif

   The policy store schema is extended. You have created the policy store schema.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

   -v

   Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

   Default value: stdout

-cf

(Optional) Imports sensitive data using FIPS-compatible cryptography.

Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
-d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.
2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.
3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.
4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.
5. Run the following command:

   XPSDDInstall FssSmObjects.xdd
6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Chapter 7: Configuring SOA Security Manager Data in a Relational Database

This section contains the following topics:

Relational Databases as a Policy or Key Store

The SOA Security Manager policy store is the repository for all policy–related information. All Policy Servers in a SOA Security Manager installation must share the same policy store data, either directly or through replication. SOA Security Manager is installed with tools that let administrators move policy store data from one storage facility to another.

When you install the Policy Server, you can automatically configure one of the following directory servers as a policy store:

- Sun Java System Directory Server Enterprise Edition (formerly Sun ONE/iPlanet)
- Microsoft ADAM
- Microsoft AD LDS

You can configure the Policy Server to use another LDAP directory server, a SQL Server database, or an Oracle database as a policy store after you have completed the Policy Server installation. Also, after installation, you can use the Policy Server Management Console to point the Policy Server to another policy store.

You can use a supported database to store SOA Security Manager policy store data. SOA Security Manager keys, audit logs, and session data can be stored in the policy store or in a separate database.

Storing keys in a separate database may be required to implement single sign–on functionality.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager Platform Support Matrix on the Technical Support site.

To locate the support matrix from the Technical Support site

1. Click Support By Product.
2. Select CA SOA Security Manager from the Select a Product list.
3. Click CA SOA Security Manager Platform Support Matrices under Product Status.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Installation Road Map

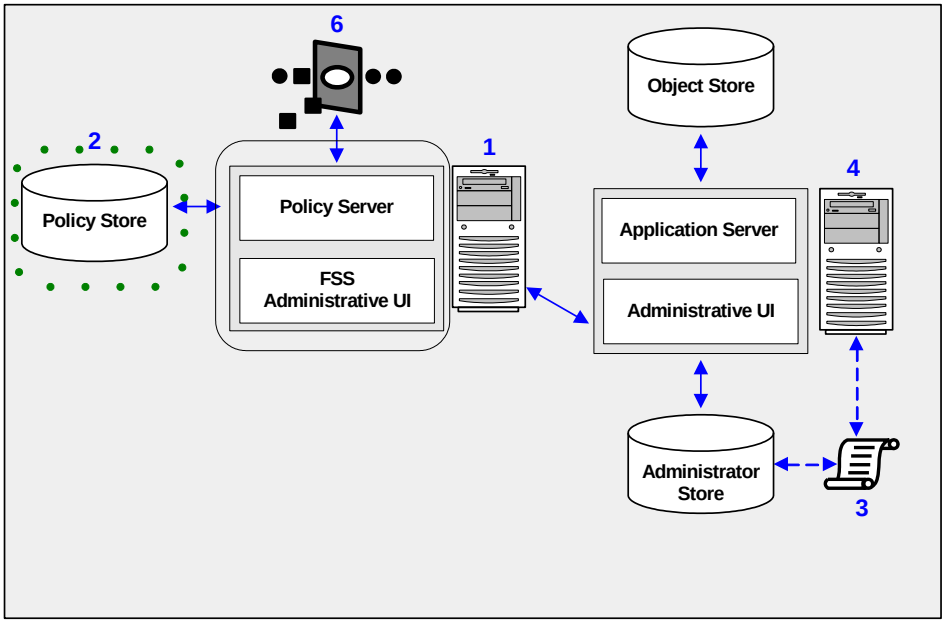
The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- The components surrounded by a solid line represent the Policy Server and a non-registered FSS Administrative UI. A Policy Server must be installed before configuring a policy store. Do not continue with the installation process if a Policy Server is not yet part of your environment.

  Note: The FSS Administrative UI is installed with the Policy Server. Unless you need to generate WS-Security SAML assertion tokens, you may safely leave the eTrust SiteMinder FSS on the Policy Server machine without registering it with the Policy Server.

- The component surrounded by a green dotted line represents the policy store, which you configure at this point in the installation process. The policy store contains all of the Policy Server data. You configure a policy store in either an LDAP or relational database.

  Note: The following figure depicts a single policy/key store instance. Although not illustrated, your environment may use separate instances for individual policy and key stores.



The following sections in the documentation detail how to:

- Configure the policy store.
- Establish a connection to the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Important Considerations

Consider the following before configuring a policy store:

- Avoid possible policy store corruption—Ensure that the database server to which the policy store is to be installed is configured to store objects in UTF-8 form:
  - (Oracle) Ensure that the database is configured to store objects in UTF-8 form. Oracle supports unicode within many of their character sets. For more information on configuring your database to store objects in UTF-8 form, see your vendor-specific documentation.
  - (MS SQL Server) Ensure that the database is configured using the default collation (SQL_Latin1_General_CP1_CI_AS). Using a collation that is case sensitive may result in unexpected behaviors. For more information on configuring your database to store objects using the default collation, refer to your vendor-specific documentation.
- Do not use brackets around the IP address when using IPv6 ODBC data sources or the connection fails.

  Example: use fec0::9255:20c:29ff:fe47:8089 instead of [fec0::9255:20c:29ff:fe47:8089]

  Note: More information on IPv6-supported databases exist in the SOA Security Manager Platform Support Matrix on the Technical Support site.

To locate the support matrix on the Support site

1. Click Technical Support.
2. Click Support By Product.
3. Select CA SOA Security Manager from the Select a Product Page list.

   Scroll to Product Status and click Platform Support Matrices.

Schema Files for Relational Databases

SOA Security Manager provides schema files to create the individual schema for storing policies; keys; logs; session data; token data, such as Encotone TeleID data; and sample users. You can store SOA Security Manager data in a single SQL Server or Oracle database, or run each script on its own to create a separate:

- policy store
- key store
- logging database
- token store
- session store
- sample users database

Note: The SOA Security Manager schema files are installed with the Policy Server. If the Policy Server is installed on a UNIX system, copy the schema files from <siteminder_home>/db/SQL directory to a temporary directory (C:\temp) on the Windows system to which the database is installed.

SQL Server Schema Files

The following SQL Server schema files are provided in the *policy_server_home*\db\SQL directory:

policy_server_home

Specifies the Policy Server installation path.

sm_mssql_ps.sql

Creates the schema for a policy store and key store.

Note: If you are storing keys in a different database, this schema file creates the schema for the key store data.

sm_mssql_logs.sql

Creates the schema for SOA Security Manager audit logs in a SQL Server database.

sm_mssql_token.sql

Creates the schema for storing token data, such as Encotone TeleID data, in a SQL Server database.

sm_mssql_ss.sql

Creates the schema for the Session Server in a SQL Server database.

smsampleusers_sqlserver.sql

Creates the schema for SOA Security Manager sample users in a SQL Server database and populates the database with sample users. For example, if you look in the script, you can see a sample user named GeorgeC with a password of siteminder.

The following SQL Server schema file is provided in *policy_server_home*\xps\db:

SQLServer.sql

Creates the XPS schema for a SQL Server policy store.

Oracle Schema Files

The following Oracle schema files are provided in the *policy_server_home*\db\SQL directory.

policy_server_home

Specifies the Policy Server installation path.

sm_oracle_ps.sql

Creates the SiteMinder policy store or key store (if you are storing keys in a different database) in an Oracle database.

sm_oracle_logs.sql

Creates the schema for SiteMinder audit logs in an Oracle database.

sm_oracle_token.sql

Creates the schema for storing token data, such as Encotone TeleID data, in an Oracle database.

sm_oracle_ss.sql

Creates the schema for the Session Server in an Oracle database.

smsampleusers_oracle.sql

Creates the schema for SiteMinder sample users in an Oracle database and populates the database with sample users. For example, if you look in the script, you can see a sample user named GeorgeC with a password of siteminder.

The following Oracle schema file is provided in the *policy_server_home*\xps\db directory.

Oracle.sql

Creates the XPS schema for an Oracle policy store.

Configure a SQL Server Policy Store

Policy Servers installed on either Windows or UNIX systems can use a single SQL Server database to function as a:

- policy store
- key store
- token store
- session store
- logging database

Using a single database simplifies administration tasks. The following sections provide instruction on how to configure a single database server to store SOA Security Manager data.

Note: The database must be installed on a Windows system. Additionally, the SOA Security Manager schema files are installed with the Policy Server. If the Policy Server is installed on a UNIX system, copy the schema files from <siteminder_home>/db/SQL directory to a temporary directory (C:\temp) on the Windows system to which the database is installed.

Gather Database Information

Configuring a single SQL Server database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Note: Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Gather the following information before configuring the policy store or any other type of SOA Security Manager data store. You can use the SQL Server Information Worksheet to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Database instance name** —Determine the name of the database instance that is to function as the policy store or data store.
- **Administrative account** —Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

- **Administrative Password** —Determine the password for the Administrative account.
- (W) **Data source name** —Determine the name you will use to identify the data source.

  Example: SM SQL Server Wire DS.
- (W) **SQL Server name** —Determine the name of the SQL Server database that contains the instance that is to function as the policy store.
- (U) **Policy Server root** —Determine the explicit path to where the Policy Server is installed.
- (U) **IP Address** —Determine the IP Address of the SQL Server database.

More information:

SQL Server Information Worksheet (see page 327)

How to Configure the Policy Store

Complete the following procedures to configure a SQL Server database as a policy store:

Note: Ensure you have gathered the required database information before beginning. Some of the following procedures require this information.

1. Make sure the SQL Server database instance that is to contain the SOA Security Manager data is accessible from the Policy Server machine.
2. Using the SQL Server Enterprise Manager, create the database instance for the SOA Security Manager data store.

   Example: smdatastore.
3. Create the SOA Security Manager Schema.
4. Configure a SQL Server Data Source for SOA Security Manager.
   - (Windows) Create a SQL Server Data Source.
   - (UNIX) Create a SQL Server Data Source on UNIX Systems.
   - (UNIX) Configure the SQL Server Wire Protocol Driver.
5. Point the Policy Server to the Database.
6. Set the SiteMinder Super User Password.
7. Import the Default SOA Security Manager Objects.
8. Import the Policy Store Data Definitions.
9. Restart the Policy Server.

Create the SOA Security Manager Schema

You create the SOA Security Manager schema so that SQL Server database can store policy, key, token session, and audit logging information.

The following warnings are displayed when running the policy store and audit logging schema files and do not affect the policy store configuration:

- Warning: The table 'smvariable5' has been created but its maximum row size (8746) exceeds the maximum number of bytes per row (8060). INSERT or UPDATE of a row in this table will fail if the resulting row length exceeds 8060 bytes.
- Warning: The table 'smodbcquery4' has been created but its maximum row size (64635) exceeds the maximum number of bytes per row (8060). INSERT or UPDATE of a row in this table will fail if the resulting row length exceeds 8060 bytes.
- Warning: The table 'smaccesslog4' has been created but its maximum row size (9668) exceeds the maximum number of bytes per row (8060). INSERT or UPDATE of a row in this table will fail if the resulting row length exceeds 8060 bytes.

To create the SOA Security Manager Schema

1. Start the Query Analyzer and log in as the person who administers the Policy Server database.
2. Select the database instance from the database list.
3. Open sm_mssql_ps.sql in a text editor and copy the contents of the entire file.
4. Paste the schema from sm_mssql_ps.sql into the query, and execute the query.

   The policy and key store schema is added to the database.
5. Open SQLServer.sql in a text editor and copy the contents of the entire file.
6. Paste the schema from SQLServer.sql into the query, and execute the query.

   The policy store schema is extended.
7. Repeat steps three and four to use the policy store as a logging database, token store, and session store.

   The respective schema files are:

   - sm_mssql_logs.sql
   - sm_mssql_token.sql
   - sm_mssql_ss.sql

   The respective SOA Security Manager schema is added to the database.

Note: You are not required to configure the policy store to store additional SOA Security Manager data. You can configure individual databases to function as one or more separate logging databases, key stores, token stores, and session stores.

The database can store SOA Security Manager data.

Configure a SQL Server Data Source for SiteMinder

If you are using ODBC, you need to configure a data source for the SQL Server wire protocol driver.

Create a SQL Server Data Source on Windows

ODBC requires that you configure a data source for the SQL Server wire protocol.

Note: This procedure only applies if the Policy Server is installed on a Windows System.

To create the data source on Windows

1. Click Start and select Programs, Administrative Tools, ODBC Data Sources.

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab.

   System data source settings appear.

3. Click Add.

   The Create New Data Source dialog appears.

4. Select SOA Security Manager SQL Server Wire Protocol and click Finish.

   The ODBC SQL Server Wire Protocol Driver Setup dialog appears.

5. Enter the data source name in the Data Source Name field.

   Example: SOA Security Manager Data Source.

   Note: Take note of your data source name. This information is required as you configure your database as a policy store.

6. Enter the name of the SQL Server host system in the Server field.

7. Enter the database name in the Database Name field.

8. Click Test.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The SQL Server data source is configured and appears in the System Data Sources list.

Create a SQL Server Data Sources on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding a SQL Server Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the Oracle or SQL drivers under [SOA Security Manager Data Source].

Again, to configure a SQL Server data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the SQL Server Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Enter the following under [ODBC Data Sources]:

   SiteMinder Data Source=DataDirect 5.3 SQL Server Wire Protocol.

3. Depending on the data source you are configuring, edit the one or more of the data source sections with the following information:

   Driver=*nete_ps_root*/odbc/lib/NSmass23.so
   Description=DataDirect 5.0 SQL Server Wire Protocol
   Database=*SiteMinder Data*
   Address=*myhost*, 1433
   QuotedId=No
   AnsiNPW=No

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path, rather than one with an environment variable.

Example: export/smuser/siteminder

SOA Security Manager Data

Specifies the SQL Server database instance name.

myhost

Specifies the IP Address of the SQL Server database.

1433

Represents the default listening port for SQL Server.

4. Save the file.

   The wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can access the SOA Security Manager data in the policy store.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Policy Store from the Database list.

4. Enter the name of the data source in the Data Source Information field.

   - **Windows** - this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX** - this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, ensure that you enter the correct value.

5. Enter and confirm the username and password of the database account that has full access rights to the database instance in the respective fields.

6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the 25 connection default for best performance.

7. Click Apply.

   The settings are saved.

8. Select Key Store from the Database list.

   Data source information appears.

9. Select the Use the policy store database check box and click Apply.

10. Select Audit Logs from the Database list.

    Data source settings appear.

11. Select the Use the policy store database check box and click Apply.

12. Select Token Data from the Database list.

    Data source settings appear.

13. Select the User policy store database check box and click Apply.

14. Click Test Connection.

    SiteMinder returns a confirmation that the Policy Server can access the data store.

15. Click OK.

    The Policy Server is configured to use the database as a policy store, key store, token store, session store, and logging database.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

   -v

   Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

   Default value: stdout

   -cf

   (Optional) Imports sensitive data using FIPS-compatible cryptography.

   Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

   smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.

3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.

4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.

5. Run the following command:

   XPSDDInstall FssSmObjects.xdd

6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Configure an Oracle Policy Store

Policy Servers installed on either Windows or UNIX systems can use a single Oracle database to function as a:

- policy store
- key store
- token store
- session store
- logging database

Using a single database simplifies administrative tasks. The following sections provide instruction on how to configure a single database server to store SOA Security Manager data.

Prerequisites for an Oracle 10g Database

After installing the Oracle 10g database, complete the following prerequisites:

- Create a table space for the policy store.
- Create a user with appropriate privileges to manage this table space in the database.

Create an Oracle 10g Table Space for the Policy Store

Creating a table space for the policy store is a prerequisite for an Oracle 10g database only.

To create an Oracle 10g table space for the policy store

1. In the Oracle Enterprise Manager 10g Database Control, log in as the SYSDBA user with appropriate privileges to manage the Oracle database.
2. On the Oracle global database's configuration screen, select Administration, Tablespaces.
3. On the Tablespaces screen, click Create.
4. On the Create Tablespaces screen, enter a table space name, and click ADD.

   Example: NETE_TB

5. On the Create Tablespaces: Add Datafile screen:
   a. Enter a file name.

      Example: NETE_TB

   b. Specify the file size.

      Example: 100 MB

   c. Click Continue.

   Oracle creates the table space and displays it on the Tablespaces screen.

Complete the prerequisites by creating a user to manage the table space for the policy store.

More Information:

Create an Oracle 10g User to Manage the Policy Store's Table Space (see page 150)

Create an Oracle 10g User to Manage the Policy Store's Table Space

Creating a user to manage table space for the policy store is a prerequisite for an Oracle 10g database only.

To create a user to manage table space for the policy store

1. On the Oracle global database's configuration screen, select Administration, Users.
2. On the Create Tablespaces screen, click Create.
3. On the Create User screen, enter the:
   - Name for the user.

     Example: NETE
   - Password for the user.
   - Default Tablespace that you created.
   - Temporary tablespace.

     Example: TEMP
4. Click Roles.
5. Select Modify.
6. On the Modify Roles screen:
   a. Select CONNECT and RESOURCE as a roles for this user.
   b. Click Apply.

7. Start sqlplus in a command window, by entering:
   a. sqlplus
   b. the credentials for the policy store user created on the Create User screen.

   You have completed the prerequisites for an Oracle 10g database, and can now configure a SOA Security Manager data store for the database.

Gather Database Information

Configuring a single Oracle database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring Oracle data source for UNIX.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

More information:

Oracle Information Worksheet (see page 328)
Oracle RAC Information Worksheet (see page 328)

Required Information

Gather the following required information before configuring a supported Oracle or Oracle RAC database as a policy store or any other type of SOA Security Manager data store:

- (U) **Policy Server installation path**—Identify the explicit path to where the Policy Server is installed.
- **Data source**—Determine the name you will use to identify the Oracle data source.

  Example: SM Oracle Server Wire DS.

- **Database administrative account**—Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

  Note: Ensure the administrative account does not have the DB role. Audit-based reports will not return correct results if the administrative account has the DB role.

- **Database administrative Password**—Determine the password for the Administrative account.

Oracle Database Information

Gather the following information only if you are configuring a supported Oracle database as a policy store or any other type of SOA Security Manager data store:

- **Oracle machine name**—Determine the name of the machine on which the Oracle database is installed.
- **Oracle instance service name**—Determine the service name of the database instance to which you will connect. The tnsnames.ora file specifies service names.
- **Oracle port number**—Determine the port number on which the Oracle database is listening.

Oracle RAC Database Information

Gather the following information only if you are configuring a supported Oracle RAC database as a policy store or any other SOA Security Manager data store:

- **Oracle RAC system service name—**Determine the service name for the entire system.

  Example: In the following tnsnames.ora file, SMDB is the service name for the entire system:

```
SMDB=
(Description =
(ADDRESS = PROTOCOL = TCP)(HOST = nete_servername1)(PORT=1521
(ADDRESS = PROTOCOL = TCP)(HOST = nete_servername2)(PORT=1521)
(ADDRESS = PROTOCOL = TCP)(HOST = nete_servername3)(PORT=1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA=
(SERVER = DEDICATED)
(SERVER_NAME = SMDB))
)
```

- **Oracle RAC node service names**—Determine the service names for each node in the system.

- **Oracle RAC node IP addresses**—Determine the IP Address of each node in the Oracle RAC system.

  Note: If you are using Oracle RAC 10g, determine the virtual IP address of each node in the system.

- **Oracle RAC node port numbers**—Determine the port number for each node in the Oracle RAC system.

How to Configure the Policy Store

To configure a single Oracle database as policy store, key store, token store, session store, and logging database, complete the following procedures:

Note: Ensure you have gathered the required database information before beginning. Some of the following procedures require this information.

1. Ensure that the Oracle database instance that is to contain the SOA Security Manager data is accessible from the Policy Server machine. Test the communication using tnsping or sqlplus.
2. Create the SOA Security Manager Schema.
3. Configure an Oracle Data Source for SOA Security Manager:
   - (Windows) Create an Oracle Data Source on Windows Systems.
   - (Windows) Create an Oracle RAC Data Source on Windows Systems.
   - (UNIX) Create an Oracle Data Source on UNIX.
   - (UNIX) Configure the Wire Protocol Driver.
   - (UNIX) Configure the Oracle Wire Protocol Driver.
   - (UNIX) Configure the Oracle RAC Wire Protocol Driver.
4. Point the Policy Server to the Database.
5. Set the SiteMinder Super User Password.
6. Import the Default SOA Security Manager Objects.
7. Import the Policy Store Data Definitions.
8. Restart the Policy Server.

Create the SOA Security Manager Schema

You create the SOA Security Manager schema so a single Oracle database can store policy, key, token, session, and audit logging information.

To create the SOA Security Manager Schema

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/sm_oracle_ps.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The policy store and key store schema is added to the database.

3. Import the following script

   $NETE_PS_ROOT/xps/Oracle.sql

   The policy store schema is extended.

4. Import the following scripts to use the policy store as a logging database, token store, and/or session store.

   The respective schema files are:

   - sm_oracle_logs.sql
   - sm_oracle_token.sql
   - sm_oracle_ss.sql

   The respective SOA Security Manager schema is created in the database.

   Note: You are not required to configure the policy store to store additional SOA Security Manager data. You can configure individual databases to function as one or more separate logging databases, key stores, token stores, and session stores.

   The database can store SOA Security Manager data.

Configure an Oracle Data Source for SOA Security Manager

If you are using ODBC, you need to configure a data source for the Oracle wire protocol driver.

Create an Oracle Data Source on Windows

To create an Oracle data source on Windows

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the machine name where the Oracle database is installed in the Host Name field.

6. Enter the port number where the Oracle database is listening on the machine in the Port Number field.

7. Enter the name of the Oracle instance to which you want to connect in the SID field.

   Note: The service name is specified in the tnsnames.ora file. The SID is the system identifier for the database instance. The tnsnames.ora file contains service names and details that Oracle uses to identify and connect to Oracle instances.

   Example: if the tnsnames.ora file contains the following entry for an Oracle instance, you enter instance1 in the SID field:

   ```
   instance1 =
       (Description=
       (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
       (Connect_DATA_ = (SID = SIDofinstance1))
       )
   ```

8. Click Test Connection.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The Oracle data source is configured for the wire protocol driver.

Create an Oracle RAC Data Source on Windows

You can configure Oracle RAC 9.2.0.6 and 10.1.0.4 instances with the Policy Server as a single data source name similar to that of a single instance Oracle database. However, the data source name for Oracle RAC is different than a regular ODBC data source.

In an Oracle RAC system, in addition to the SID or ServiceName for each Oracle instance, there is also a ServiceName for the entire Oracle RAC system. When configuring a data source, configure the data source name to use this ServiceName when connecting to an Oracle RAC.

To configure an Oracle RAC data source

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the IP Address of the first node in the Oracle RAC system in the Host field.

   Oracle RAC 10g: Enter the virtual IP Address.

6. Enter the service name for the entire Oracle RAC system in the Service Name field.

   Example: In the following tnsnames.ora file, the SMDB value is the service name for the entire Oracle RAC system, which contains 3 nodes:

```
SMDB=
      (Description =
(ADDRESS = (Protocol = TCP)(HOST = nete_servername1)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername2)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername3)(PORT = 1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = SMDB)
)
```

7. Click the Failover tab.

   Failover settings appear.

8. Specify the host name or virtual IP Address, port number, and service name for the remaining Oracle RAC nodes in the environment in the Alternate Servers field.

   Note: The ServiceName is the service name for the entire Oracle RAC system.

9. Specify the AlternateServers to provide connection failover to the other Oracle nodes if the primary server is not accepting connections. The entry should have the following format:

   (HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

10. Select LoadBalancing.
11. Click OK

    The Oracle RAC data source is configured for the wire protocol driver.

Create an Oracle Data Source on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding an Oracle Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the SQL Server or Oracle drivers under [SOA Security Manager Data Source].

Again, to configure an Oracle data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the Oracle Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:

Driver=*nete_ps_root*/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
LoginID=*uid*
Password=*pwd*
HostName=*nete_servername*
PortNumber=1521
SID=*nete_serverid*
CatalogOptions=0
ProcedureResults=0
EnableDisableParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies the explicit path to where the Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servername

Specifies the machine name where the Oracle database is installed.

nete_serverid

Specifies the Oracle instance service name (SID). The SID is the system identifier for the database instance.

Example: In the following sample tnsnames.ora file, the value instance1 is the SID.

```
instance1 =
(Description =
(ADDRESS = (Protocol = TCP)(Host = myhost)(Port = 1521)
(CONNECT_DATA = (SID = instance1))
)
```

3. Save the file.

The Oracle wire protocol driver is configured.

Configure the Oracle RAC Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:
   - Add ServiceName=nete_servicename
   - Add AlternateServers=
   - Add Loadbalancing=1
   - Remove or comment SID=nete_serverid

The modified text for the data source should appear as follows:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
Logon=uid
Password=pwd
HostName=nete_servername1
PortNumber=1521
ServiceName=nete_servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
AlternateServers=
LoadBalancing=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path to the directory where Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servicename1

Specifies the IP Address of the first Oracle RAC node.

(Oracle 10g) Specifies the virtual IP Address of the first Oracle RAC node.

nete_servicename

Specifies the service name for the entire Oracle RAC system.

AlternateServers

Specifies the connection failover to the other Oracle nodes, if the primary server is not accepting connections.

Example:
(HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

LoadBalancing=1

Turns on client load balancing, which helps to distribute new connections to keep RAC nodes from being overwhelmed with connection requests. When enabled, the order in which primary and alternate database servers are accessed is random.

3. Save the file.

   The Oracle wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can access the SOA Security Manager data in the policy store.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Policy Store from the Database list.

4. Enter the name of the data source in the Data Source Information field.

   - **Windows** - this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX** - this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, ensure that you enter the correct value.

5. Enter and confirm the username and password of the database account that has full access rights to the database instance in the respective fields.

6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the 25 connection default for best performance.

7. Click Apply.

   The settings are saved.

8. Select Key Store from the Database list.

   Data source information appears.

9. Select the Use the policy store database check box and click Apply.

10. Select Audit Logs from the Database list.

    Data source settings appear.

11. Select the Use the policy store database check box and click Apply.

12. Select Token Data from the Database list.

    Data source settings appear.

13. Select the User policy store database check box and click Apply.

14. Click Test Connection.

    SiteMinder returns a confirmation that the Policy Server can access the data store.

15. Click OK.

    The Policy Server is configured to use the database as a policy store, key store, token store, session store, and logging database.

Set the SOA Security Manager Super User Password

The Policy Server installer installs the FSS Administrative UI and SOA Security Manager utilities with a default Super User account named SOA Security Manager. You set the SOA Security Manager Super User password to let an administrator:

- log into the FSS Administrative UI using the default SOA Security Manager account
- use SOA Security Manager utilities using the default account

Note: You use the smreg utility to change the SOA Security Manager Super User. The smreg utility is located at the top level of the Policy Server installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To set the SOA Security Manager Super User password

1. Copy smreg to *policy_server_home*\bin.

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   smreg -su *siteminder_super_user_password*

   siteminder_super_user_password

   Specifies the SOA Security Manager Super User password.

   Limit: The password can be from 6 to 24 characters in length.

   Note: The password is case-insensitive, except in cases where the password is stored in an Oracle policy store.

3. Delete smreg from *policy_server_home*\bin.

   Note: Deleting smreg prevents anyone from changing the password without knowing the previous one.

   The password for the default SOA Security Manager Super User account is set.

Import the Default Policy Store Objects

Importing the default policy store objects sets up the policy store for use with the Administrative UI. The default policy store objects are required to store policy information in the policy store.

Note: If you have installed the Policy Server in FIPS-only mode, ensure you use the -cf argument when importing the default policy store objects.

To import the default policy store objects

1. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\smpolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -v

   If the argument contains spaces, use double quotes around the entire argument.

   Windows example: smobjimport
   -i"C:\Program Files\CA\siteminder\db\smdif\smpolicy.smdif" -d"SM Admin" -wPassword -v

   UNIX example: smobjimport
   -i$NETE_PS_ROOT/db/smdif/smpolicy.smdif -d"SM Admin" -wPassword -v

   policy_server_home

   Specifies the Policy Server installation path.

   -d*siteminder_super_user_name*

   Specifies the name of the SOA Security Manager super user account.

   -w*siteminder_super_user_password*

   Specifies the password for the SOA Security Manager super user account.

   -v

   Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

   Default value: stdout

   -cf

   (Optional) Imports sensitive data using FIPS-compatible cryptography.

   Note: This argument is only required if the Policy Server is operating in FIPS-only mode.

   smobjimport imports the policy store objects. The objects are automatically imported to the appropriate locations.

2. Run the following command:

   smobjimport -i*policy_server_home*\db\smdif\ampolicy.smdif
   -d*siteminder_super_user_name* -w*siteminder_super_user_password* -f -v -l -c

policy_server_home

Specifies the Policy Server installation path.

-d*siteminder_super_user_name*

Specifies the name of the SOA Security Manager super user account.

-w*siteminder_super_user_password*

Specifies the password for the SOA Security Manager super user account.

-f

Overrides duplicate objects.

-v

Turns on tracing and outputs error, warning, and comment messages in verbose format so that you can monitor the status of the import.

Default value: stdout

-l

Creates a log file.

-c

Indicates that the smdif input file contains unencrypted data.

smobjimport imports the policy store objects. These objects are automatically imported to the appropriate locations.

Note: Importing ampolicy.smdif makes available eTrust SiteMinder FSS, Web Service Variables, and eTelligent Rules functionality that is separately licensed from SOA Security Manager. If you intend on using the latter functionality, contact your CA account representative for more information about licensing.

You can now import the policy store data definitions.

Import the Policy Store Data Definitions

Importing the policy store data definitions is required to use the policy store with the Administrative UI. The base definitions describe the policy store data.

To import the base policy store objects

1. Open a command window and navigate to one of the following locations:
   - **Windows**—*policy_server_home*\xps\dd
   - **UNIX**—*policy_server_home*/xps/dd

   policy_server_home

   Specifies the Policy Server installation path.

2. Run the following command:

   Important! Run the XPSDDInstall tool with the data definition files in the following order or the imports fail.

   XPSDDInstall SmObjects.xdd

   XPSDDInstall imports the required data definitions.

3. Run the following command:

   XPSDDInstall EPMObjects.xdd

   XPSDDInstall imports the required data definitions.

4. Run the following command:

   XPSDDInstall SecCat.xdd

   XPSDDInstall imports the required data definitions.

5. Run the following command:

   XPSDDInstall FssSmObjects.xdd

6. Run the following command:

   XPSDDInstall SoaSmObjects.xdd

   You have imported all of the required policy store data definitions.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Configure SQL Server Data Stores

SOA Security Manager key, audit log, token, and session information can each be stored in a separate database.

The following sections detail how to configure individual data stores.

Note: Storing keys in a separate database may be required to implement single sign-on functionality. More information on key management exists in the *Policy Server Administration Guide*.

How to Store Key Information in SQL Server

To configure a SQL Server database as a standalone key store, complete the following procedures:

1. Gather Database Information.
2. Create the Key Store Schema.
3. Configure a SQL Server Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single SQL Server database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Note: Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Gather the following information before configuring the policy store or any other type of SOA Security Manager data store. You can use the SQL Server Information Worksheet to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Database instance name** —Determine the name of the database instance that is to function as the policy store or data store.
- **Administrative account** —Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.
- **Administrative Password** —Determine the password for the Administrative account.

- (W) **Data source name** —Determine the name you will use to identify the data source.

  Example: SM SQL Server Wire DS.

- (W) **SQL Server name** —Determine the name of the SQL Server database that contains the instance that is to function as the policy store.
- (U) **Policy Server root** —Determine the explicit path to where the Policy Server is installed.
- (U) **IP Address** —Determine the IP Address of the SQL Server database.

More information:

SQL Server Information Worksheet (see page 327)

Create the Key Store Schema

You create the key store schema so the SQL Server database can store key information.

To create the key store schema

1. Open sm_mssql_ps.sql in a text editor and copy the contents of the entire file.
2. Start the Query Analyzer and log in as the who administers the Policy Server database.
3. Select the database instance from the database list.
4. Paste the schema from sm_mssql_ps.sql into the query.
5. Execute the query.

   The SOA Security Manager key store schema is created in the database.

Configure a SQL Server Data Source for SiteMinder

If you are using ODBC, you need to configure a data source for the SQL Server wire protocol driver.

Create a SQL Server Data Source on Windows

ODBC requires that you configure a data source for the SQL Server wire protocol.

Note: This procedure only applies if the Policy Server is installed on a Windows System.

To create the data source on Windows

1. Click Start and select Programs, Administrative Tools, ODBC Data Sources.

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab.

   System data source settings appear.

3. Click Add.

   The Create New Data Source dialog appears.

4. Select SOA Security Manager SQL Server Wire Protocol and click Finish.

   The ODBC SQL Server Wire Protocol Driver Setup dialog appears.

5. Enter the data source name in the Data Source Name field.

   Example: SOA Security Manager Data Source.

   Note: Take note of your data source name. This information is required as you configure your database as a policy store.

6. Enter the name of the SQL Server host system in the Server field.

7. Enter the database name in the Database Name field.

8. Click Test.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The SQL Server data source is configured and appears in the System Data Sources list.

Create a SQL Server Data Sources on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding a SQL Server Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the Oracle or SQL drivers under [SOA Security Manager Data Source].

Again, to configure a SQL Server data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the SQL Server Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Enter the following under [ODBC Data Sources]:

   SiteMinder Data Source=DataDirect 5.3 SQL Server Wire Protocol.

3. Depending on the data source you are configuring, edit the one or more of the data source sections with the following information:

   Driver=*nete_ps_root*/odbc/lib/NSmass23.so
   Description=DataDirect 5.0 SQL Server Wire Protocol
   Database=*SiteMinder Data*
   Address=*myhost*, 1433
   QuotedId=No
   AnsiNPW=No

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

   nete_ps_root

   Specifies an explicit path, rather than one with an environment variable.

   Example: export/smuser/siteminder

   SOA Security Manager Data

   Specifies the SQL Server database instance name.

   myhost

   Specifies the IP Address of the SQL Server database.

   1433

   Represents the default listening port for SQL Server.

4. Save the file.

   The wire protocol driver is configured.

Point the Policy Server to Database

You point the Policy Server to the database so the Policy Server can read and store key information.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Key Store from the Database list and clear the Use Policy Store database check box.

   Data source settings become active.

4. Enter the name of the data source in the Data Source Information field.

   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.

5. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.

6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.

7. Click Apply.

   The settings are saved.

8. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.

9. Click OK.

   The Policy Server is configured to use the database as a key store

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Audit Logs in SQL Server

To configure a SQL Server database as a standalone audit logging database, complete the following procedures:

1. Gather Database Information.
2. Create the Audit Log Schema.
3. Configure a SQL Server Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single SQL Server database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Note: Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Gather the following information before configuring the policy store or any other type of SOA Security Manager data store. You can use the SQL Server Information Worksheet to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Database instance name** —Determine the name of the database instance that is to function as the policy store or data store.
- **Administrative account** —Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.
- **Administrative Password** —Determine the password for the Administrative account.
- (W) **Data source name** —Determine the name you will use to identify the data source.

  Example: SM SQL Server Wire DS.
- (W) **SQL Server name** —Determine the name of the SQL Server database that contains the instance that is to function as the policy store.
- (U) **Policy Server root** —Determine the explicit path to where the Policy Server is installed.
- (U) **IP Address** —Determine the IP Address of the SQL Server database.

More information:

SQL Server Information Worksheet (see page 327)

Create the Audit Log Schema

You create the logging schema so the SQL Server database can store audit logs.

To create the audit log schema

1. Open sm_mssql_logs.sql in a text editor and copy the contents of the entire file.
2. Start the Query Analyzer and log in as the user who administers the Policy Server database.
3. Select the database instance from the database list.

4. Paste the schema from sm_mssql_logs.sql into the query.
5. Execute the query.

   The SOA Security Manager audit log store schema is created in the database.

Configure a SQL Server Data Source for SiteMinder

If you are using ODBC, you need to configure a data source for the SQL Server wire protocol driver.

Create a SQL Server Data Source on Windows

ODBC requires that you configure a data source for the SQL Server wire protocol.

Note: This procedure only applies if the Policy Server is installed on a Windows System.

To create the data source on Windows

1. Click Start and select Programs, Administrative Tools, ODBC Data Sources.

   The ODBC Data Source Administrator appears.
2. Click the System DSN tab.

   System data source settings appear.
3. Click Add.

   The Create New Data Source dialog appears.
4. Select SOA Security Manager SQL Server Wire Protocol and click Finish.

   The ODBC SQL Server Wire Protocol Driver Setup dialog appears.
5. Enter the data source name in the Data Source Name field.

   Example: SOA Security Manager Data Source.

   Note: Take note of your data source name. This information is required as you configure your database as a policy store.
6. Enter the name of the SQL Server host system in the Server field.
7. Enter the database name in the Database Name field.
8. Click Test.

   The connection settings are tested and a prompt appears specifying that the connection is successful.
9. Click OK.

   The SQL Server data source is configured and appears in the System Data Sources list.

Create a SQL Server Data Sources on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding a SQL Server Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the Oracle or SQL drivers under [SOA Security Manager Data Source].

Again, to configure a SQL Server data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the SQL Server Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Enter the following under [ODBC Data Sources]:

   SiteMinder Data Source=DataDirect 5.3 SQL Server Wire Protocol.

3. Depending on the data source you are configuring, edit the one or more of the data source sections with the following information:

   Driver=*nete_ps_root*/odbc/lib/NSmass23.so
   Description=DataDirect 5.0 SQL Server Wire Protocol
   Database=*SiteMinder Data*
   Address=*myhost*, 1433
   QuotedId=No
   AnsiNPW=No

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path, rather than one with an environment variable.

Example: export/smuser/siteminder

SOA Security Manager Data

Specifies the SQL Server database instance name.

myhost

Specifies the IP Address of the SQL Server database.

1433

Represents the default listening port for SQL Server.

4. Save the file.

   The wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can read and store audit logs.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Audit Logs from the Database list.

4. Select ODBC from the Storage list.

   Data source settings become active.

5. Enter the name of the data source in the Data Source Information field.

   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.

6. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.

7. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.

8. Click Apply.

   The settings are saved.

9. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.

10. Click OK.

    The Policy Server is configured to use the database as an audit logging database.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Token Data in SQL Server

To configure a SQL Server database as a standalone token store, complete the following procedures:

1. Gather Database Information.
2. Create the Token Store Schema.
3. Configure a SQL Server Data Source for SOA Security Manager.

4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single SQL Server database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Note: Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Gather the following information before configuring the policy store or any other type of SOA Security Manager data store. You can use the SQL Server Information Worksheet to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Database instance name** —Determine the name of the database instance that is to function as the policy store or data store.
- **Administrative account** —Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.
- **Administrative Password** —Determine the password for the Administrative account.
- (W) **Data source name** —Determine the name you will use to identify the data source.

  Example: SM SQL Server Wire DS.
- (W) **SQL Server name** —Determine the name of the SQL Server database that contains the instance that is to function as the policy store.
- (U) **Policy Server root** —Determine the explicit path to where the Policy Server is installed.
- (U) **IP Address** —Determine the IP Address of the SQL Server database.

More information:

SQL Server Information Worksheet (see page 327)

Create the Token Store Schema

You create the logging schema so the SQL Server database can store token information.

To create the token store schema

1. Open sm_mssql_token.sql in a text editor and copy the contents of the entire file.
2. Start the Query Analyzer and log in as the user who administers the Policy Server database.
3. Select the database instance from the database list.
4. Paste the schema from sm_mssql_token.sql into the query.
5. Execute the query.

   The SOA Security Manager token store schema is created in the database.

Configure a SQL Server Data Source for SiteMinder

If you are using ODBC, you need to configure a data source for the SQL Server wire protocol driver.

Create a SQL Server Data Source on Windows

ODBC requires that you configure a data source for the SQL Server wire protocol.

Note: This procedure only applies if the Policy Server is installed on a Windows System.

To create the data source on Windows

1. Click Start and select Programs, Administrative Tools, ODBC Data Sources.

   The ODBC Data Source Administrator appears.
2. Click the System DSN tab.

   System data source settings appear.
3. Click Add.

   The Create New Data Source dialog appears.
4. Select SOA Security Manager SQL Server Wire Protocol and click Finish.

   The ODBC SQL Server Wire Protocol Driver Setup dialog appears.
5. Enter the data source name in the Data Source Name field.

   Example: SOA Security Manager Data Source.

   Note: Take note of your data source name. This information is required as you configure your database as a policy store.

6. Enter the name of the SQL Server host system in the Server field.
7. Enter the database name in the Database Name field.
8. Click Test.

   The connection settings are tested and a prompt appears specifying that the connection is successful.
9. Click OK.

   The SQL Server data source is configured and appears in the System Data Sources list.

Create a SQL Server Data Sources on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding a SQL Server Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the Oracle or SQL drivers under [SOA Security Manager Data Source].

Again, to configure a SQL Server data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the SQL Server Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Enter the following under [ODBC Data Sources]:

   SiteMinder Data Source=DataDirect 5.3 SQL Server Wire Protocol.

3. Depending on the data source you are configuring, edit the one or more of the data source sections with the following information:

   Driver=*nete_ps_root*/odbc/lib/NSmass23.so
   Description=DataDirect 5.0 SQL Server Wire Protocol
   Database=*SiteMinder Data*
   Address=*myhost*, 1433
   QuotedId=No
   AnsiNPW=No

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

   nete_ps_root

   Specifies an explicit path, rather than one with an environment variable.

   Example: export/smuser/siteminder

   SOA Security Manager Data

   Specifies the SQL Server database instance name.

   myhost

   Specifies the IP Address of the SQL Server database.

   1433

   Represents the default listening port for SQL Server.

4. Save the file.

   The wire protocol driver is configured.

Point the Policy Store to the Database

You point the Policy Server to the database so the Policy Server can read and store audit logs.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Token Data from the Database list.

   Data source settings become active.

4. Enter the name of the data source in the Data Source Information field.
   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.
5. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.
6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.
7. Click Apply.

   The settings are saved.
8. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.
9. Click OK.

   The Policy Server is configured to use the database as a token store.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Session Information in SQL Server

To configure a SQL Server database as a session store, complete the following procedures:

1. Gather Database Information.
2. Create the Session Store Schema.
3. Configure a SQL Server Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single SQL Server database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Note: Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Gather the following information before configuring the policy store or any other type of SOA Security Manager data store. You can use the SQL Server Information Worksheet to record your values.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

- **Database instance name** —Determine the name of the database instance that is to function as the policy store or data store.
- **Administrative account** —Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.
- **Administrative Password** —Determine the password for the Administrative account.
- (W) **Data source name** —Determine the name you will use to identify the data source.

  Example: SM SQL Server Wire DS.

- (W) **SQL Server name** —Determine the name of the SQL Server database that contains the instance that is to function as the policy store.
- (U) **Policy Server root** —Determine the explicit path to where the Policy Server is installed.
- (U) **IP Address** —Determine the IP Address of the SQL Server database.

More information:

SQL Server Information Worksheet (see page 327)

Create the Session Store Schema

You create the session store schema so the SQL Server database can store and read session information.

To create the session store schema

1. Open sm_mssql_ss.sql in a text editor and copy the contents of the entire file.
2. Start the Query Analyzer and log in as the who administers the Policy Server database.
3. Select the database instance from the database list.
4. Paste the schema from sm_mssql_ss.sql into the query.
5. Execute the query.

   The session store schema is created in the database.

Configure a SQL Server Data Source for SiteMinder

If you are using ODBC, you need to configure a data source for the SQL Server wire protocol driver.

Create a SQL Server Data Source on Windows

ODBC requires that you configure a data source for the SQL Server wire protocol.

Note: This procedure only applies if the Policy Server is installed on a Windows System.

To create the data source on Windows

1. Click Start and select Programs, Administrative Tools, ODBC Data Sources.

   The ODBC Data Source Administrator appears.
2. Click the System DSN tab.

   System data source settings appear.

3. Click Add.

   The Create New Data Source dialog appears.

4. Select SOA Security Manager SQL Server Wire Protocol and click Finish.

   The ODBC SQL Server Wire Protocol Driver Setup dialog appears.

5. Enter the data source name in the Data Source Name field.

   Example: SOA Security Manager Data Source.

   Note: Take note of your data source name. This information is required as you configure your database as a policy store.

6. Enter the name of the SQL Server host system in the Server field.

7. Enter the database name in the Database Name field.

8. Click Test.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The SQL Server data source is configured and appears in the System Data Sources list.

Create a SQL Server Data Sources on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding a SQL Server Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the Oracle or SQL drivers under [SOA Security Manager Data Source].

Again, to configure a SQL Server data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename sqlserverwire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the SQL Server Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Enter the following under [ODBC Data Sources]:

   SiteMinder Data Source=DataDirect 5.3 SQL Server Wire Protocol

3. Depending on the data source you are configuring, edit the one or more of the data source sections with the following information:

   Driver=*nete_ps_root*/odbc/lib/NSmass23.so
   Description=DataDirect 5.0 SQL Server Wire Protocol
   Database=*SiteMinder Data*
   Address=*myhost*, 1433
   QuotedId=No
   AnsiNPW=No

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

   nete_ps_root

   Specifies an explicit path, rather than one with an environment variable.

   Example: export/smuser/siteminder

   SOA Security Manager Data

   Specifies the SQL Server database instance name.

   myhost

   Specifies the IP Address of the SQL Server database.

   1433

   Represents the default listening port for SQL Server.

4. Save the file.

   The wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can read and store session information.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select Session Server from the Database list.

   Data source settings become active.

3. Enter the name of the data source in the Data Source Information field.
   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.
4. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.
5. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.
6. Click Apply.

   The settings are saved.
7. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.
8. Click OK.

   The Policy Server is configured to use the database as a session store.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Configure Oracle Data Stores

SOA Security Manager key, audit log, token, and session information can each be stored in a separate database.

The following sections detail how to configure individual data stores.

Note: Storing keys in a separate database may be required to implement single sign-on functionality. More information on key management exists in the *Policy Server Administration Guide*.

How to Store Key Information in Oracle

To configure an Oracle database as key store, complete the following procedures:

1. Gather Database Information.
2. Create the Key Store Schema.
3. Configure an Oracle Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single Oracle database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring Oracle data source for UNIX.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

More information:

Oracle Information Worksheet (see page 328)
Oracle RAC Information Worksheet (see page 328)

Required Information

Gather the following required information before configuring a supported Oracle or Oracle RAC database as a policy store or any other type of SOA Security Manager data store:

- (U) **Policy Server installation path**—Identify the explicit path to where the Policy Server is installed.
- **Data source**—Determine the name you will use to identify the Oracle data source.

  Example: SM Oracle Server Wire DS.
- **Database administrative account**—Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

  Note: Ensure the administrative account does not have the DB role. Audit-based reports will not return correct results if the administrative account has the DB role.
- **Database administrative Password**—Determine the password for the Administrative account.

Oracle Database Information

Gather the following information only if you are configuring a supported Oracle database as a policy store or any other type of SOA Security Manager data store:

- **Oracle machine name**—Determine the name of the machine on which the Oracle database is installed.
- **Oracle instance service name**—Determine the service name of the database instance to which you will connect. The tnsnames.ora file specifies service names.
- **Oracle port number**—Determine the port number on which the Oracle database is listening.

Oracle RAC Database Information

Gather the following information only if you are configuring a supported Oracle RAC database as a policy store or any other SOA Security Manager data store:

- **Oracle RAC system service name—**Determine the service name for the entire system.

  Example: In the following tnsnames.ora file, SMDB is the service name for the entire system:

  ```
  SMDB=
  (Description =
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername1)(PORT=1521
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername2)(PORT=1521)
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername3)(PORT=1521))
  (LOAD_BALANCE = yes)
  (CONNECT_DATA=
  (SERVER = DEDICATED)
  (SERVER_NAME = SMDB))
  )
  ```

- **Oracle RAC node service names**—Determine the service names for each node in the system.
- **Oracle RAC node IP addresses**—Determine the IP Address of each node in the Oracle RAC system.

  Note: If you are using Oracle RAC 10g, determine the virtual IP address of each node in the system.
- **Oracle RAC node port numbers**—Determine the port number for each node in the Oracle RAC system.

Create the Key Store Schema

You create the key store schema so the Oracle database can store key information.

To create the SOA Security Manager schema

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/sm_oracle_ps.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The key store schema is created in the database.

Configure an Oracle Data Source for SOA Security Manager

If you are using ODBC, you need to configure a data source for the Oracle wire protocol driver.

Create an Oracle Data Source on Windows

To create an Oracle data source on Windows

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the machine name where the Oracle database is installed in the Host Name field.

6. Enter the port number where the Oracle database is listening on the machine in the Port Number field.

7. Enter the name of the Oracle instance to which you want to connect in the SID field.

   Note: The service name is specified in the tnsnames.ora file. The SID is the system identifier for the database instance. The tnsnames.ora file contains service names and details that Oracle uses to identify and connect to Oracle instances.

   Example: if the tnsnames.ora file contains the following entry for an Oracle instance, you enter instance1 in the SID field:

   ```
   instance1 =
       (Description=
       (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
       (Connect_DATA_ = (SID = SIDofinstance1))
       )
   ```

8. Click Test Connection.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The Oracle data source is configured for the wire protocol driver.

Create an Oracle RAC Data Source on Windows

You can configure Oracle RAC 9.2.0.6 and 10.1.0.4 instances with the Policy Server as a single data source name similar to that of a single instance Oracle database. However, the data source name for Oracle RAC is different than a regular ODBC data source.

In an Oracle RAC system, in addition to the SID or ServiceName for each Oracle instance, there is also a ServiceName for the entire Oracle RAC system. When configuring a data source, configure the data source name to use this ServiceName when connecting to an Oracle RAC.

To configure an Oracle RAC data source

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the IP Address of the first node in the Oracle RAC system in the Host field.

   Oracle RAC 10g: Enter the virtual IP Address.

6. Enter the service name for the entire Oracle RAC system in the Service Name field.

   Example: In the following tnsnames.ora file, the SMDB value is the service name for the entire Oracle RAC system, which contains 3 nodes:

```
SMDB=
    (Description =
(ADDRESS = (Protocol = TCP)(HOST = nete_servername1)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername2)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername3)(PORT = 1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = SMDB)
)
```

7. Click the Failover tab.

   Failover settings appear.

8. Specify the host name or virtual IP Address, port number, and service name for the remaining Oracle RAC nodes in the environment in the Alternate Servers field.

   Note: The ServiceName is the service name for the entire Oracle RAC system.

9. Specify the AlternateServers to provide connection failover to the other Oracle nodes if the primary server is not accepting connections. The entry should have the following format:

   (HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

10. Select LoadBalancing.

11. Click OK

    The Oracle RAC data source is configured for the wire protocol driver.

Create an Oracle Data Source on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding an Oracle Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the SQL Server or Oracle drivers under [SOA Security Manager Data Source].

Again, to configure an Oracle data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the Oracle Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
LoginID=uid
Password=pwd
HostName=nete_servername
PortNumber=1521
SID=nete_serverid
CatalogOptions=0
ProcedureResults=0
EnableDisableParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies the explicit path to where the Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servername

Specifies the machine name where the Oracle database is installed.

nete_serverid

Specifies the Oracle instance service name (SID). The SID is the system identifier for the database instance.

Example: In the following sample tnsnames.ora file, the value instance1 is the SID.

```
instance1 =
(Description =
(ADDRESS = (Protocol = TCP)(Host = myhost)(Port = 1521)
(CONNECT_DATA = (SID = instance1))
)
```

3. Save the file.

   The Oracle wire protocol driver is configured.

Configure the Oracle RAC Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:
   - Add ServiceName=nete_servicename
   - Add AlternateServers=
   - Add Loadbalancing=1
   - Remove or comment SID=nete_serverid

The modified text for the data source should appear as follows:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
Logon=uid
Password=pwd
HostName=nete_servername1
PortNumber=1521
ServiceName=nete_servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
AlternateServers=
LoadBalancing=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path to the directory where Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servicename1

Specifies the IP Address of the first Oracle RAC node.

(Oracle 10g) Specifies the virtual IP Address of the first Oracle RAC node.

nete_servicename

Specifies the service name for the entire Oracle RAC system.

AlternateServers

Specifies the connection failover to the other Oracle nodes, if the primary server is not accepting connections.

Example:
(HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

LoadBalancing=1

Turns on client load balancing, which helps to distribute new connections to keep RAC nodes from being overwhelmed with connection requests. When enabled, the order in which primary and alternate database servers are accessed is random.

3. Save the file.

   The Oracle wire protocol driver is configured.

Point the Policy Server to Database

You point the Policy Server to the database so the Policy Server can read and store key information.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Key Store from the Database list and clear the Use Policy Store database check box.

   Data source settings become active.

4. Enter the name of the data source in the Data Source Information field.

   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.

5. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.

6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.

7. Click Apply.

   The settings are saved.

8. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.

9. Click OK.

   The Policy Server is configured to use the database as a key store

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Audit Logs in Oracle

To configure an Oracle database as key store, complete the following procedures:

1. Gather Database Information.
2. Create the Audit Log Schema.
3. Configure an Oracle Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single Oracle database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring Oracle data source for UNIX.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

More information:

Oracle Information Worksheet (see page 328)
Oracle RAC Information Worksheet (see page 328)

Required Information

Gather the following required information before configuring a supported Oracle or Oracle RAC database as a policy store or any other type of SOA Security Manager data store:

- (U) **Policy Server installation path**—Identify the explicit path to where the Policy Server is installed.
- **Data source**—Determine the name you will use to identify the Oracle data source.

  Example: SM Oracle Server Wire DS.
- **Database administrative account**—Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

  Note: Ensure the administrative account does not have the DB role. Audit-based reports will not return correct results if the administrative account has the DB role.
- **Database administrative Password**—Determine the password for the Administrative account.

Oracle Database Information

Gather the following information only if you are configuring a supported Oracle database as a policy store or any other type of SOA Security Manager data store:

- **Oracle machine name**—Determine the name of the machine on which the Oracle database is installed.
- **Oracle instance service name**—Determine the service name of the database instance to which you will connect. The tnsnames.ora file specifies service names.
- **Oracle port number**—Determine the port number on which the Oracle database is listening.

Oracle RAC Database Information

Gather the following information only if you are configuring a supported Oracle RAC database as a policy store or any other SOA Security Manager data store:

- **Oracle RAC system service name—**Determine the service name for the entire system.

  Example: In the following tnsnames.ora file, SMDB is the service name for the entire system:

  ```
  SMDB=
  (Description =
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername1)(PORT=1521
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername2)(PORT=1521)
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername3)(PORT=1521))
  (LOAD_BALANCE = yes)
  (CONNECT_DATA=
  (SERVER = DEDICATED)
  (SERVER_NAME = SMDB))
  )
  ```

- **Oracle RAC node service names**—Determine the service names for each node in the system.
- **Oracle RAC node IP addresses**—Determine the IP Address of each node in the Oracle RAC system.

  Note: If you are using Oracle RAC 10g, determine the virtual IP address of each node in the system.
- **Oracle RAC node port numbers**—Determine the port number for each node in the Oracle RAC system.

Create the Audit Log Schema

You create the audit log schema so the Oracle database can store audit logs.

To create the SOA Security Manager schema

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/sm_oracle_logs.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The audit log schema is created in the database.

Configure an Oracle Data Source for SOA Security Manager

If you are using ODBC, you need to configure a data source for the Oracle wire protocol driver.

Create an Oracle Data Source on Windows

To create an Oracle data source on Windows

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the machine name where the Oracle database is installed in the Host Name field.

6. Enter the port number where the Oracle database is listening on the machine in the Port Number field.

7. Enter the name of the Oracle instance to which you want to connect in the SID field.

   Note: The service name is specified in the tnsnames.ora file. The SID is the system identifier for the database instance. The tnsnames.ora file contains service names and details that Oracle uses to identify and connect to Oracle instances.

   Example: if the tnsnames.ora file contains the following entry for an Oracle instance, you enter instance1 in the SID field:

   ```
   instance1 =
       (Description=
       (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
       (Connect_DATA_ = (SID = SIDofinstance1))
       )
   ```

8. Click Test Connection.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The Oracle data source is configured for the wire protocol driver.

Create an Oracle RAC Data Source on Windows

You can configure Oracle RAC 9.2.0.6 and 10.1.0.4 instances with the Policy Server as a single data source name similar to that of a single instance Oracle database. However, the data source name for Oracle RAC is different than a regular ODBC data source.

In an Oracle RAC system, in addition to the SID or ServiceName for each Oracle instance, there is also a ServiceName for the entire Oracle RAC system. When configuring a data source, configure the data source name to use this ServiceName when connecting to an Oracle RAC.

To configure an Oracle RAC data source

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the IP Address of the first node in the Oracle RAC system in the Host field.

   Oracle RAC 10g: Enter the virtual IP Address.

6. Enter the service name for the entire Oracle RAC system in the Service Name field.

   Example: In the following tnsnames.ora file, the SMDB value is the service name for the entire Oracle RAC system, which contains 3 nodes:

```
SMDB=
    (Description =
(ADDRESS = (Protocol = TCP)(HOST = nete_servername1)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername2)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername3)(PORT = 1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = SMDB)
)
```

7. Click the Failover tab.

   Failover settings appear.

8. Specify the host name or virtual IP Address, port number, and service name for the remaining Oracle RAC nodes in the environment in the Alternate Servers field.

   Note: The ServiceName is the service name for the entire Oracle RAC system.

9. Specify the AlternateServers to provide connection failover to the other Oracle nodes if the primary server is not accepting connections. The entry should have the following format:

   (HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

10. Select LoadBalancing.

11. Click OK

    The Oracle RAC data source is configured for the wire protocol driver.

Create an Oracle Data Source on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding an Oracle Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the SQL Server or Oracle drivers under [SOA Security Manager Data Source].

Again, to configure an Oracle data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the Oracle Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:

Driver=*nete_ps_root*/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
LoginID=*uid*
Password=*pwd*
HostName=*nete_servername*
PortNumber=1521
SID=*nete_serverid*
CatalogOptions=0
ProcedureResults=0
EnableDisableParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies the explicit path to where the Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servername

Specifies the machine name where the Oracle database is installed.

nete_serverid

Specifies the Oracle instance service name (SID). The SID is the system identifier for the database instance.

Example: In the following sample tnsnames.ora file, the value instance1 is the SID.

```
instance1 =
(Description =
(ADDRESS = (Protocol = TCP)(Host = myhost)(Port = 1521)
(CONNECT_DATA = (SID = instance1))
)
```

3. Save the file.

   The Oracle wire protocol driver is configured.

Configure the Oracle RAC Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:
   - Add ServiceName=nete_servicename
   - Add AlternateServers=
   - Add Loadbalancing=1
   - Remove or comment SID=nete_serverid

The modified text for the data source should appear as follows:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
Logon=uid
Password=pwd
HostName=nete_servername1
PortNumber=1521
ServiceName=nete_servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
AlternateServers=
LoadBalancing=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path to the directory where Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servicename1

Specifies the IP Address of the first Oracle RAC node.

(Oracle 10g) Specifies the virtual IP Address of the first Oracle RAC node.

nete_servicename

Specifies the service name for the entire Oracle RAC system.

AlternateServers

Specifies the connection failover to the other Oracle nodes, if the primary server is not accepting connections.

Example:
(HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

LoadBalancing=1

Turns on client load balancing, which helps to distribute new connections to keep RAC nodes from being overwhelmed with connection requests. When enabled, the order in which primary and alternate database servers are accessed is random.

3. Save the file.

   The Oracle wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can read and store audit logs.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Audit Logs from the Database list.
4. Select ODBC from the Storage list.

   Data source settings become active.

5. Enter the name of the data source in the Data Source Information field.
   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.
6. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.

7. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.

8. Click Apply.

   The settings are saved.

9. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.

10. Click OK.

    The Policy Server is configured to use the database as an audit logging database.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Token Information in Oracle

To configure an Oracle database as a token store, complete the following procedures:

1. Gather Database Information.
2. Create the Token Store Schema.
3. Configure an Oracle Data Source for SOA Security Manager.

4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single Oracle database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring Oracle data source for UNIX.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

More information:

Oracle Information Worksheet (see page 328)
Oracle RAC Information Worksheet (see page 328)

Required Information

Gather the following required information before configuring a supported Oracle or Oracle RAC database as a policy store or any other type of SOA Security Manager data store:

- (U) **Policy Server installation path**—Identify the explicit path to where the Policy Server is installed.
- **Data source**—Determine the name you will use to identify the Oracle data source.

  Example: SM Oracle Server Wire DS.
- **Database administrative account**—Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

  Note: Ensure the administrative account does not have the DB role. Audit-based reports will not return correct results if the administrative account has the DB role.
- **Database administrative Password**—Determine the password for the Administrative account.

Oracle Database Information

Gather the following information only if you are configuring a supported Oracle database as a policy store or any other type of SOA Security Manager data store:

- **Oracle machine name**—Determine the name of the machine on which the Oracle database is installed.
- **Oracle instance service name**—Determine the service name of the database instance to which you will connect. The tnsnames.ora file specifies service names.
- **Oracle port number**—Determine the port number on which the Oracle database is listening.

Oracle RAC Database Information

Gather the following information only if you are configuring a supported Oracle RAC database as a policy store or any other SOA Security Manager data store:

- **Oracle RAC system service name—**Determine the service name for the entire system.

  Example: In the following tnsnames.ora file, SMDB is the service name for the entire system:

  ```
  SMDB=
  (Description =
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername1)(PORT=1521
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername2)(PORT=1521)
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername3)(PORT=1521))
  (LOAD_BALANCE = yes)
  (CONNECT_DATA=
  (SERVER = DEDICATED)
  (SERVER_NAME = SMDB))
  )
  ```

- **Oracle RAC node service names**—Determine the service names for each node in the system.
- **Oracle RAC node IP addresses**—Determine the IP Address of each node in the Oracle RAC system.

  Note: If you are using Oracle RAC 10g, determine the virtual IP address of each node in the system.
- **Oracle RAC node port numbers**—Determine the port number for each node in the Oracle RAC system.

Create the Token Store Schema

You create the token store schema so the Oracle database can store token information.

To create the SOA Security Manager schema

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/sm_oracle_token.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The token store schema is created in the database.

Configure an Oracle Data Source for SOA Security Manager

If you are using ODBC, you need to configure a data source for the Oracle wire protocol driver.

Create an Oracle Data Source on Windows

To create an Oracle data source on Windows

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the machine name where the Oracle database is installed in the Host Name field.

6. Enter the port number where the Oracle database is listening on the machine in the Port Number field.

7. Enter the name of the Oracle instance to which you want to connect in the SID field.

   Note: The service name is specified in the tnsnames.ora file. The SID is the system identifier for the database instance. The tnsnames.ora file contains service names and details that Oracle uses to identify and connect to Oracle instances.

   Example: if the tnsnames.ora file contains the following entry for an Oracle instance, you enter instance1 in the SID field:

   ```
   instance1 =
       (Description=
       (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
       (Connect_DATA_ = (SID = SIDofinstance1))
       )
   ```

8. Click Test Connection.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The Oracle data source is configured for the wire protocol driver.

Create an Oracle RAC Data Source on Windows

You can configure Oracle RAC 9.2.0.6 and 10.1.0.4 instances with the Policy Server as a single data source name similar to that of a single instance Oracle database. However, the data source name for Oracle RAC is different than a regular ODBC data source.

In an Oracle RAC system, in addition to the SID or ServiceName for each Oracle instance, there is also a ServiceName for the entire Oracle RAC system. When configuring a data source, configure the data source name to use this ServiceName when connecting to an Oracle RAC.

To configure an Oracle RAC data source

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the IP Address of the first node in the Oracle RAC system in the Host field.

   Oracle RAC 10g: Enter the virtual IP Address.

6. Enter the service name for the entire Oracle RAC system in the Service Name field.

   Example: In the following tnsnames.ora file, the SMDB value is the service name for the entire Oracle RAC system, which contains 3 nodes:

```
SMDB=
    (Description =
(ADDRESS = (Protocol = TCP)(HOST = nete_servername1)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername2)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername3)(PORT = 1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = SMDB)
)
```

7. Click the Failover tab.

   Failover settings appear.

8. Specify the host name or virtual IP Address, port number, and service name for the remaining Oracle RAC nodes in the environment in the Alternate Servers field.

   Note: The ServiceName is the service name for the entire Oracle RAC system.

9. Specify the AlternateServers to provide connection failover to the other Oracle nodes if the primary server is not accepting connections. The entry should have the following format:

   (HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

10. Select LoadBalancing.

11. Click OK

    The Oracle RAC data source is configured for the wire protocol driver.

Create an Oracle Data Source on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding an Oracle Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the SQL Server or Oracle drivers under [SOA Security Manager Data Source].

Again, to configure an Oracle data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the Oracle Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
LoginID=uid
Password=pwd
HostName=nete_servername
PortNumber=1521
SID=nete_serverid
CatalogOptions=0
ProcedureResults=0
EnableDisableParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies the explicit path to where the Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servername

Specifies the machine name where the Oracle database is installed.

nete_serverid

Specifies the Oracle instance service name (SID). The SID is the system identifier for the database instance.

Example: In the following sample tnsnames.ora file, the value instance1 is the SID.

```
instance1 =
(Description =
(ADDRESS = (Protocol = TCP)(Host = myhost)(Port = 1521)
(CONNECT_DATA = (SID = instance1))
)
```

3. Save the file.

   The Oracle wire protocol driver is configured.

Configure the Oracle RAC Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:
   - Add ServiceName=nete_servicename
   - Add AlternateServers=
   - Add Loadbalancing=1
   - Remove or comment SID=nete_serverid

The modified text for the data source should appear as follows:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
Logon=uid
Password=pwd
HostName=nete_servername1
PortNumber=1521
ServiceName=nete_servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
AlternateServers=
LoadBalancing=1
```

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies an explicit path to the directory where Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servicename1

Specifies the IP Address of the first Oracle RAC node.

(Oracle 10g) Specifies the virtual IP Address of the first Oracle RAC node.

nete_servicename

Specifies the service name for the entire Oracle RAC system.

AlternateServers

Specifies the connection failover to the other Oracle nodes, if the primary server is not accepting connections.

Example:
(HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

LoadBalancing=1

Turns on client load balancing, which helps to distribute new connections to keep RAC nodes from being overwhelmed with connection requests. When enabled, the order in which primary and alternate database servers are accessed is random.

3. Save the file.

   The Oracle wire protocol driver is configured.

Point the Policy Store to the Database

You point the Policy Server to the database so the Policy Server can read and store audit logs.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select ODBC from the Storage list.

   ODBC settings appear.

3. Select Token Data from the Database list.

   Data source settings become active.

4. Enter the name of the data source in the Data Source Information field.

   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.

5. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.

6. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.

7. Click Apply.

   The settings are saved.

8. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.

9. Click OK.

   The Policy Server is configured to use the database as a token store.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.

3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

How to Store Session Information in Oracle

To configure an Oracle database as a session store, complete the following procedures:

1. Gather Database Information.
2. Create the Session Store Schema.
3. Configure an Oracle Data Source for SOA Security Manager.
4. Point the Policy Server to the Database.
5. Restart the Policy Server.

Note: If you are trying to configure or upgrade a SOA Security Manager store listed in the SOA Security Manager Platform Support Matrix and cannot find the procedures in this guide, see the *Directory Configuration Guide*.

Gather Database Information

Configuring a single Oracle database to function as a policy store or any other type of SOA Security Manager data store requires specific database information.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring Oracle data source for UNIX.

Note: Policy and data store worksheets are provided to help you gather and record information before configuring or upgrading a SOA Security Manager data store. You can print the applicable worksheet and use it to record required information before beginning.

More information:

Oracle Information Worksheet (see page 328)
Oracle RAC Information Worksheet (see page 328)

Required Information

Gather the following required information before configuring a supported Oracle or Oracle RAC database as a policy store or any other type of SOA Security Manager data store:

- (U) **Policy Server installation path**—Identify the explicit path to where the Policy Server is installed.
- **Data source**—Determine the name you will use to identify the Oracle data source.

  Example: SM Oracle Server Wire DS.
- **Database administrative account**—Determine the user name of an account with privileges to create, read, modify, and delete objects in the database.

  Note: Ensure the administrative account does not have the DB role. Audit-based reports will not return correct results if the administrative account has the DB role.
- **Database administrative Password**—Determine the password for the Administrative account.

Oracle Database Information

Gather the following information only if you are configuring a supported Oracle database as a policy store or any other type of SOA Security Manager data store:

- **Oracle machine name**—Determine the name of the machine on which the Oracle database is installed.
- **Oracle instance service name**—Determine the service name of the database instance to which you will connect. The tnsnames.ora file specifies service names.
- **Oracle port number**—Determine the port number on which the Oracle database is listening.

Oracle RAC Database Information

Gather the following information only if you are configuring a supported Oracle RAC database as a policy store or any other SOA Security Manager data store:

- **Oracle RAC system service name—**Determine the service name for the entire system.

  Example: In the following tnsnames.ora file, SMDB is the service name for the entire system:

  ```
  SMDB=
  (Description =
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername1)(PORT=1521
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername2)(PORT=1521)
  (ADDRESS = PROTOCOL = TCP)(HOST = nete_servername3)(PORT=1521))
  (LOAD_BALANCE = yes)
  (CONNECT_DATA=
  (SERVER = DEDICATED)
  (SERVER_NAME = SMDB))
  )
  ```

- **Oracle RAC node service names**—Determine the service names for each node in the system.
- **Oracle RAC node IP addresses**—Determine the IP Address of each node in the Oracle RAC system.

  Note: If you are using Oracle RAC 10g, determine the virtual IP address of each node in the system.
- **Oracle RAC node port numbers**—Determine the port number for each node in the Oracle RAC system.

Create the Session Store Schema

You create the session store schema so the Oracle database can store session information.

To create the SOA Security Manager schema

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/sm_oracle_ss.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The session store schema is created in the database.

Configure an Oracle Data Source for SOA Security Manager

If you are using ODBC, you need to configure a data source for the Oracle wire protocol driver.

Create an Oracle Data Source on Windows

To create an Oracle data source on Windows

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the machine name where the Oracle database is installed in the Host Name field.

6. Enter the port number where the Oracle database is listening on the machine in the Port Number field.

7. Enter the name of the Oracle instance to which you want to connect in the SID field.

   Note: The service name is specified in the tnsnames.ora file. The SID is the system identifier for the database instance. The tnsnames.ora file contains service names and details that Oracle uses to identify and connect to Oracle instances.

Example: if the tnsnames.ora file contains the following entry for an Oracle instance, you enter instance1 in the SID field:

```
instance1 =
    (Description=
    (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
    (Connect_DATA_ = (SID = SIDofinstance1))
    )
```

8. Click Test Connection.

   The connection settings are tested and a prompt appears specifying that the connection is successful.

9. Click OK.

   The Oracle data source is configured for the wire protocol driver.

Create an Oracle RAC Data Source on Windows

You can configure Oracle RAC 9.2.0.6 and 10.1.0.4 instances with the Policy Server as a single data source name similar to that of a single instance Oracle database. However, the data source name for Oracle RAC is different than a regular ODBC data source.

In an Oracle RAC system, in addition to the SID or ServiceName for each Oracle instance, there is also a ServiceName for the entire Oracle RAC system. When configuring a data source, configure the data source name to use this ServiceName when connecting to an Oracle RAC.

To configure an Oracle RAC data source

1. Select Start, Programs, Administrative Tools, Data Sources (ODBC).

   The ODBC Data Source Administrator appears.

2. Click the System DSN tab, and then click Add.

3. Select SOA Security Manager Oracle Wire Protocol, and click Finish.

   The ODBC Oracle Wire Protocol Driver Setup dialog appears. The General tab is pulled to the front.

4. Enter a name that identifies the data source in the Data Source Name field.

   Note: Record this name. You will need the data source name when pointing the Policy Server to the database.

5. Enter the IP Address of the first node in the Oracle RAC system in the Host field.

   Oracle RAC 10g: Enter the virtual IP Address.

6. Enter the service name for the entire Oracle RAC system in the Service Name field.

   Example: In the following tnsnames.ora file, the SMDB value is the service name for the entire Oracle RAC system, which contains 3 nodes:

```
SMDB=
      (Description =
(ADDRESS = (Protocol = TCP)(HOST = nete_servername1)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername2)(PORT = 1521))
(ADDRESS = (Protocol = TCP)(HOST = nete_servername3)(PORT = 1521))
(LOAD_BALANCE = yes)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = SMDB)
)
```

7. Click the Failover tab.

   Failover settings appear.

8. Specify the host name or virtual IP Address, port number, and service name for the remaining Oracle RAC nodes in the environment in the Alternate Servers field.

   Note: The ServiceName is the service name for the entire Oracle RAC system.

9. Specify the AlternateServers to provide connection failover to the other Oracle nodes if the primary server is not accepting connections. The entry should have the following format:

   (HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

10. Select LoadBalancing.

11. Click OK

    The Oracle RAC data source is configured for the wire protocol driver.

Create an Oracle Data Source on UNIX Systems

The SOA Security Manager ODBC data sources are configured using a system_odbc.ini file, which you create by renaming oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini. This system_odbc.ini file contains all of the names of the available ODBC data sources as well as the attributes that are associated with these data sources. This file must be customized to work for each site. Also, you can add additional data sources to this file, such as defining additional ODBC user directories for SOA Security Manager.

The first section of the system_odbc.ini file, [ODBC Data Sources], contains a list of all of the currently available data sources. The name before the "=" refers to a subsequent section of the file describing each individual data source. After the "=" is a comment field.

Note: If you modify of the first line of data source entry, which is [SOA Security Manager Data Source], take note of the change because you will need this value when configure your ODBC database as a policy store.

Each data source has a section in the system_odbc.ini file describing its attributes. The first attribute is the ODBC driver to be loaded when this data source is used by SOA Security Manager. The remaining attributes are specific to the driver.

Adding an Oracle Data source involves adding a new data source name in the [ODBC Data Sources] section of the file, and adding a section that describes the data source using the same name as the data source. You need to change the system_odbc.ini file if you create a new service name or want to use a different driver. You should have entries for the SQL Server or Oracle drivers under [SOA Security Manager Data Source].

Again, to configure an Oracle data source, you must first create a system_odbc.ini file in the *policy_server_installation*/db directory. To do this, you need to rename oraclewire.ini, located in *policy_server_installation*/db, to system_odbc.ini.

Configure the Oracle Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:

Driver=*nete_ps_root*/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
LoginID=*uid*
Password=*pwd*
HostName=*nete_servername*
PortNumber=1521
SID=*nete_serverid*
CatalogOptions=0
ProcedureResults=0
EnableDisableParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1

Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

nete_ps_root

Specifies the explicit path to where the Policy Server is installed.

uid

Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servername

Specifies the machine name where the Oracle database is installed.

nete_serverid

Specifies the Oracle instance service name (SID). The SID is the system identifier for the database instance.

Example: In the following sample tnsnames.ora file, the value instance1 is the SID.

```
instance1 =
(Description =
(ADDRESS = (Protocol = TCP)(Host = myhost)(Port = 1521)
(CONNECT_DATA = (SID = instance1))
)
```

3. Save the file.

   The Oracle wire protocol driver is configured.

Configure the Oracle RAC Wire Protocol Driver

You configure the wire protocol driver to specify the settings SOA Security Manager should use to connect to the database.

Note: This procedure only applies if the Policy Server is installed on a UNIX system. If you have not already done so, copy and rename oraclewire.ini to system_odbc.ini, which is located in <*policy_server_installation*>/db.

The system_odbc.ini file contains the following sections. The sections you edit are determined by the data source you are configuring:

[SOA Security Manager Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the policy store.

[SOA Security Manager Tokens Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the token store.

[SOA Security Manager Logs Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the audit log database.

[SOA Security Manager Keys Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the key store.

[SOA Security Manager Session Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the session store.

[SmSampleUsers Data Source]

Specifies the settings SOA Security Manager should use to connect to the database functioning as the sample user data store.

To configure the wire protocol driver

1. Open the system_odbc.ini file.
2. Depending on the data source you are configuring, edit the applicable data source sections with the following information:
   - Add ServiceName=nete_servicename
   - Add AlternateServers=
   - Add Loadbalancing=1
   - Remove or comment SID=nete_serverid

   The modified text for the data source should appear as follows:

```
Driver=nete_ps_root/odbc/lib/NSora23.so
Description=DataDirect 5.0 Oracle Wire Protocol
Logon=uid
Password=pwd
HostName=nete_servername1
PortNumber=1521
ServiceName=nete_servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
AlternateServers=
LoadBalancing=1
```

   Note: When editing data source information, do not use the pound sign (#). Entering a pound sign comments the information, which truncates the value. The truncated value may cause ODBC connections to fail.

   nete_ps_root

   Specifies an explicit path to the directory where Policy Server is installed.

   uid

   Specifies the user name of the database account that has full access rights to the database.

pwd

Specifies the password for the database account that has full access rights to the database.

nete_servicename1

Specifies the IP Address of the first Oracle RAC node.

(Oracle 10g) Specifies the virtual IP Address of the first Oracle RAC node.

nete_servicename

Specifies the service name for the entire Oracle RAC system.

AlternateServers

Specifies the connection failover to the other Oracle nodes, if the primary server is not accepting connections.

Example:
(HostName=nete_servername2:PortNumber=1521:ServiceName=nete_servicename[,...])

LoadBalancing=1

Turns on client load balancing, which helps to distribute new connections to keep RAC nodes from being overwhelmed with connection requests. When enabled, the order in which primary and alternate database servers are accessed is random.

3. Save the file.

   The Oracle wire protocol driver is configured.

Point the Policy Server to the Database

You point the Policy Server to the database so the Policy Server can read and store session information.

To point the Policy Server to the data store

1. Open the Policy Server Management Console, and click the Data tab.

   Database settings appear.

2. Select Session Server from the Database list.

   Data source settings become active.

3. Enter the name of the data source in the Data Source Information field.
   - **Windows—**this entry must match the name you entered in the Data Source field of the ODBC Oracle Wire Protocol Driver Setup dialog when you created the Oracle data source or the name you entered in the Data Source field of the ODBC SQL Server Wire Protocol Driver Setup dialog when you created the SQL data source.
   - **UNIX—**this entry must match the first line of the data source entry in the system_odbc.ini file. By default, the first line in the file is [SOA Security Manager Data Sources]. If you modified the first entry, be sure that you enter the correct value.
4. Enter and confirm the user name and password of the database account that has full access rights to the database instance in the respective fields.
5. Specify the maximum number of database connections allocated to SOA Security Manager.

   Note: We recommend retaining the default for best performance.
6. Click Apply.

   The settings are saved.
7. Click Test Connection.

   SiteMinder returns a confirmation that the Policy Server can access the data store.
8. Click OK.

   The Policy Server is configured to use the database as a session store.

Restart the Policy Server

You restart the Policy Server for the policy store or other data store settings to take effect.

Note: UNIX systems can use the stop-all followed by the start-all commands to restart the Policy Server.

To restart the Policy Server

1. Open the Policy Server Management Console.
2. Click the Status tab, and click Stop in the Policy Server group box.

   The Policy Server stops as indicated by the red stoplight.
3. Click Start.

   The Policy Server starts as indicated by the green stoplight.

Sample User Directories

SOA Security Manager provides schema files that populate a user directory with sample users.

The following schema files are provided in the *<siteminder_home>*\db\SQL directory:

smsampleusers_sqlserver.sql

Creates the schema for SiteMinder sample users in a SQL Server database and populates the database with sample users. For example, if you look in the script, you can see a sample user named GeorgeC with a password of siteminder.

smsampleusers_oracle.sql

Creates the schema for SiteMinder sample users in an Oracle database and populates the database with sample users. For example, if you look in the script, you can see a sample user named GeorgeC with a password of siteminder.

Note: Creating a sample user directory is optional. More information on configuring user directory connections to the Policy Server exists in the *Policy Server Configuration Guide*.

Configure an Oracle Sample User Directory

You configure a sample user directory to populate a database with sample users.

To configure the sample user directory

1. Log into Oracle with sqlplus or some other Oracle utility as the user who administers the Policy Server database information.

   Note: We recommend that you do not create SOA Security Manager schema with the SYS or SYSTEM users. If necessary, create an Oracle user, such as SMOWNER, and create the schema with that user.

2. Import the following script:

   $NETE_PS_ROOT/db/sql/smsampleusers_oracle.sql

   Note: Environment variables may not function in Oracle's SQL utility. If you experience problems importing the script using the utility, specify an explicit path.

   The user directory is populated with the sample users.

3. Configure the user directory connection to the Policy Server.

Note: More information on configuring user directory connections exists in the *Policy Server Configuration Guide*.

Configure a SQL Server Sample User Directory

You configure a sample user directory to populate a database with sample users.

To configure the sample user directory

1. Open smsampleusers_sqlserver.sql in a text editor and copy the contents of the entire file.
2. Start the Query Analyzer and log in as the user who administers the Policy Server database.
3. Select the database instance from the database list.
4. Paste the schema from smsampleusers_sqlserver.sql into the query.
5. Execute the query.

   The user directory is populated with the sample users.
6. Configure the user directory connection to the Policy Server.

Note: For more information about configuring user directory connections, see the *Policy Server Configuration Guide*.

Chapter 8: Installing the Administrative UI

This section contains the following topics:

Installation Road Map

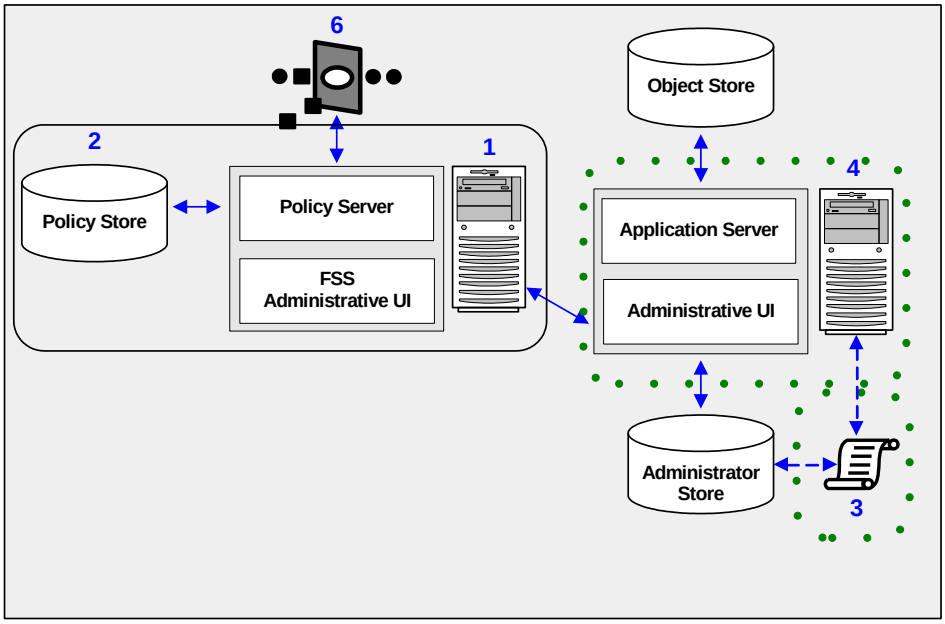
The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- The components surrounded by a solid line represent the Policy Server, policy store, and a non-registered FSS Administrative UI. A Policy Server must be installed and a policy store must be configured to register the Administrative UI. Do not continue with the installation process if a Policy Server and a policy store are not yet part of your environment.

  Note: The FSS Administrative UI is installed with the Policy Server. Unless you need to generate WS-Security SAML assertion tokens, you may safely leave the eTrust SiteMinder FSS on the Policy Server machine without registering it with the Policy Server.

- The components surrounded a green dotted line represent the directory XML file and the Administrative UI, which you configure and install at this point in the installation process.

  Note: Once you have installed and registered the Administrative UI, you can prepare for a SOA Agent installation, as illustrated by step 6.



The following sections in the documentation detail how to:

- Configure a directory XML file.
- Install the Administrative UI, which includes configuring a directory XML file, an administrator user store, and an object store.

- Register the Administrative UI with the Policy Server.
- Prepare for a SOA Agent installation.

Administrative UI Pre-Installation Checklist

You may want to print the following to use as a checklist to help ensure you meet the required system, JDK, application server, object store database, and administrative user store requirements before installing the Administrative UI.

- ☐ Confirm that the Windows or UNIX system that is to host the Administrative UI meets the minimum system requirements. Refer to Administrative User Interface System Requirements.
- ☐ Confirm that a supported application server is installed on the system that is to host the Administrative UI. Refer to:
  - JBoss as an Application Server (see page 23)
  - WebLogic as an Application Server (see page 23)
  - WebSphere as an Application Server (see page 24)
- ☐ Confirm that the required JDK is installed on the system to which the application server is installed.
- ☐ Ensure that you are using a supported SQL Server or Oracle database for the object store.

  Important! If you are using WebSphere 6.0.x as an application server and MS SQL Server 2005 for the object store, enable XA transactions in MS SQL Server 2005 before installing the Administrative UI or the installation fails. More information exists in How to Enable XA on MS SQL 2005 (see page 351).

- ☐ Ensure that all of your SOA Security Manager administrative users are stored in a single directory server or database. A single administrator user store is a requirement of the Administrative UI.

  Note: If you are installing the Administrative UI as part of an upgrade, you can use an existing user store as an administrator user store.

- ☐ Configure a directory configuration file (directory.xml). A directory configuration file describes how users are stored in the administrator user store. You supply the location of your directory configuration file during the Administrative UI installation. The Administrative UI uses the file to locate SOA Security Manager administrators. Refer to:

  - Configure an LDAP Directory Configuration File
  - Configure an ODBC Directory Configuration File

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site. This matrix also lists the Policy Server components supported on a Japanese operating system.

To locate the support matrix from the Support site

1. Click Technical Support.
2. Click Support By Product or Solution.
3. Select CA SOA Security Manager from the Select a Product or Solution Page list.
4. Click Platform Support Matrices in the Product Status group box.

You can download the latest JDK and JRE versions at the Sun Developer Network.

Configure an LDAP Directory Configuration File

A directory configuration file (directory.xml) describes how users are stored in a directory server. The Administrative UI uses the directory configuration file to locate SOA Security Manager administrators in the directory server you have designated as your administrator user store.

The directory configuration file contains one or more of the following sections:

Directory Information

Contains information about the directory configuration file that is used by the Administrative UI.

Note: Do not modify information in this section.

Provider Information

Describes the administrator user store that the Administrative UI is to use.

User Object

Describes how administrative users are stored in the user store.

To create a directory configuration file, modify one of the supplied templates. A template is provided for each of the supported LDAP directory servers.

How to Configure a Directory Configuration File

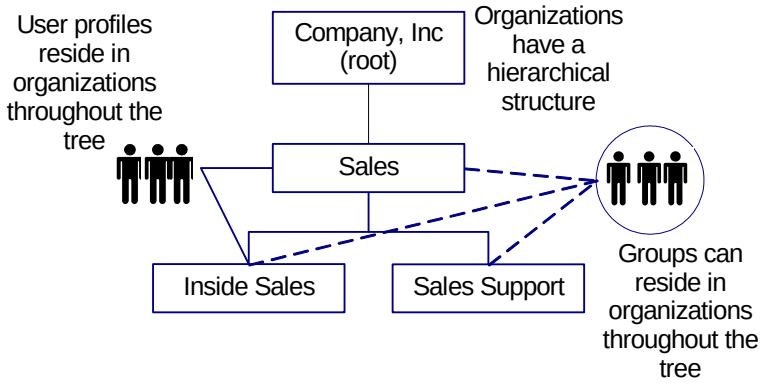
Configuring a directory XML file for an LDAP user store involves the following steps:

1. Determining the directory structure.
2. Describing the user objects in the user store by modifying a directory configuration file (directory.xml)

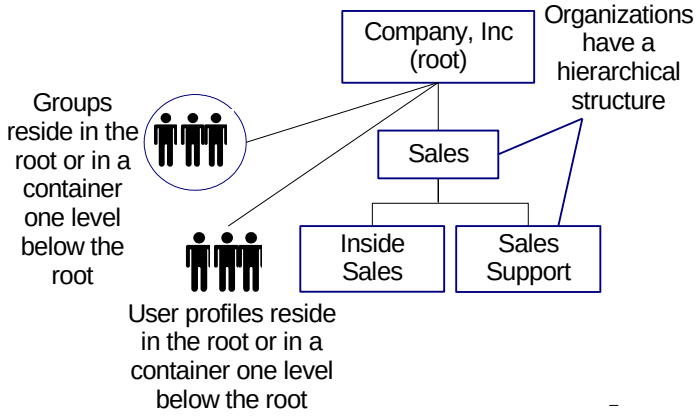
Directory Structure

The Administrative UI supports the following directory structures:

- **Hierarchical**—Contains a parent organization (root) and suborganizations. The suborganizations may also have suborganizations, which creates a multi-level structure, as shown in the following illustration:

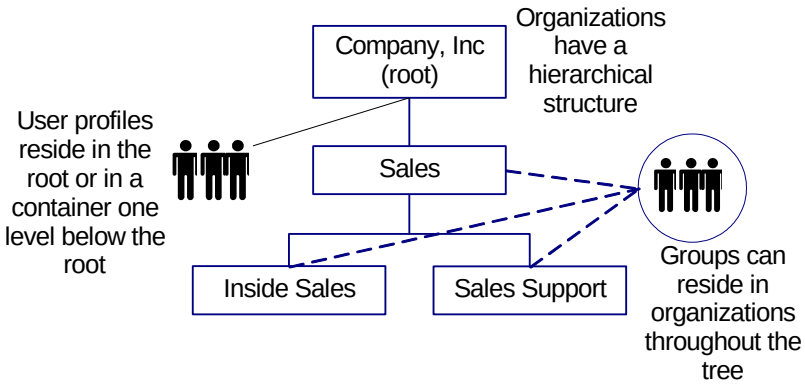


- **Flat**—User and groups are stored at the search root or in a container one level below the search root. Organizations have a hierarchical structure, as shown in the following illustration of a flat directory structure:



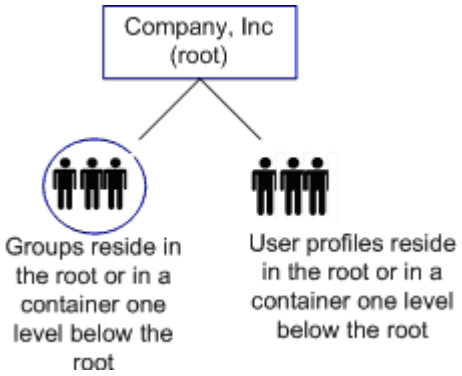
To facilitate user management and delegation in flat directory structures, users and groups belong to logical organizations. The logical organization is stored as an attribute in user and group profiles.

- **Flat User**—Organizations and groups are stored hierarchically, but users are stored at the search root or in a container one level below the search root, as shown in the following illustration of a flat user directory structure:



  In flat user directory structures, users belong to logical organizations. A user's logical organization is stored as an attribute in a user's profile.

- **No organizations**—The directory does not include organizations. Users and groups are stored at the search root or in a container one level below the search root. A no-organizations directory structure is shown in the following illustration:



Note: A directory may contain more than one type of structure. For example, user profiles may be stored in a flat structure in one part of the directory and hierarchically in another. SOA Security Manager administrators cannot reside in both structures. All administrators must reside in a single type of structure.

Select a Directory Configuration Template

SOA Security Manager supplies directory configuration templates that support different directory types and structures. To create a directory configuration file, modify the template that most closely matches your directory structure.

The templates described in the following table are located in the DirectoryConfigurationSamples folder at the top level of the Administrative UI installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

The types of directories and the corresponding configuration templates are shown in the following table:

Directory Type	Template
Active Directory (ADSI) LDAP directory with a hierarchical structure	ActiveDirectory\directory.xml
IBM Directory Server directory with a hierarchical structure	IBMDirectoryServer\directory.xml
Novell eDirectory user directory with a hierarchical structure	eDirectory\directory.xml
Oracle Internet Directory with a hierarchical structure	OracleInternetDirectory\directory.xml
Sun Java System (SunOne or iPlanet) Directory Server with a hierarchical structure	IPlanetHierarchical\directory.xml
Sun Java System (SunOne or iPlanet) Directory Server with a flat structure	IPlanetFlat\directory.xml
Sun Java System (SunOne or iPlanet) Directory Server that does not include organizations	IPlanetNoOrganizations\directory.xml
CA Directory user store with a hierarchical structure	eTrustDirectory\directory.xml
Custom directory	Use the template that most closely resembles your directory

Copy the configuration template to a new directory or save it with a different name to prevent overwriting it.

Describe an Administrator User Store

To manage an administrator user directory, the Administrative UI must understand the directory's structure and content. To describe the directory to the Administrative UI, modify the directory configuration file (directory.xml) in the appropriate template directory.

The directory configuration file has the following important conventions:

- ##—Indicates required values.

  To provide all the required information, locate all double pound signs (##) and replace them with appropriate values. For example, ##DISABLED_STATE indicates that you must supply an attribute to store the status of a user's account.

- @—Indicates values that the Administrative UI installer populates. Do not modify these values in the directory configuration file. The Administrative UI installer prompts you for this information when you install the Administrative UI. The installer writes the information to the directory configuration file during the installation.

Before you modify the directory configuration file, you need the following information:

- LDAP object classes for the user objects
- List of attributes in user profiles

How to Modify the Directory Configuration File

Perform the following steps to modify the directory configuration file.

1. Modify the default user objects (see page 255).
2. Change the default attribute descriptions (see page 255).
3. Modify well-known attributes (see page 259).

   Well-known attributes identify special attributes, such as the password attribute.

4. Configure your directory structure.

Provider Element

Configuration information is stored in the Provider element and its subelements in the directory.xml file.

The Provider element includes the following subelements:

LDAP

Describes the administrator user directory to which you are connecting.

Credentials

Provides the user name and password for accessing the LDAP user store.

Connection

Supplies the host name and port for the computer where the user store is located.

A completed Provider element may resemble the following:

```
<Provider type="LDAP" userdirectory="@SMDirName">
  <LDAP searchroot="@SMDirSearchRoot" secure="@SMDirSecure" />
  <Credentials user="@SMDirUser"
    cleartext="true">@SMDirPassword</Credentials>
  <Connection host="@SMDirHost" port="@SMDirPort" />
  </Provider>
```

The Provider element includes the following parameters:

type

Specifies the type of the database. For all LDAP user stores, specify LDAP (default).

userdirectory

Specifies the name of the administrator user store directory connection.

Note: Do not specify a name for the user directory connection in the directory.xml file. The Administrative UI installer prompts you to supply the name when you install the Administrative UI.

Note: The parameters are optional.

LDAP Subelement

The LDAP subelement includes the following parameters:

searchroot

Specifies the location in an LDAP directory that serves as the starting point for the directory—typically, an organization (o) or organizational unit (ou).

Note: The Administrative UI installer prompts you for the administrator user store search root when you install the Administrative UI. Do not manually edit this value.

secure

Forces a Secure Sockets Layer (SSL) connection to the LDAP user directory, as follows:

- True—the Administrative UI uses a secure connection.
- False—the Administrative UI connects to the user directory without SSL (default).

Note: This parameter is optional.

Credentials Subelement

To connect to an LDAP directory, the Administrative UI must provide valid credentials. The credentials are defined in the Credentials subelement, which resembles the following:

```
<Credentials user="@SMDirUser" cleartext="true">
 "MyPassword"
</Credentials>
```

The Credentials subelement includes the following parameters:

user

Specifies the login ID for an account that can access the directory.

Note: Do not specify a value for the user parameter in the directory.xml file. The Administrative UI installer prompts you to supply the login ID when you install the Administrative UI.

cleartext

Determines whether the password is displayed in clear text in the directory.xml file, as follows:

- True—The password is displayed in clear text.
- False—The password is encrypted (default).

Note: This parameter is optional.

Connection Subelement

The Connection subelement describes the location of the administrator user store that the Administrative UI is to use to identify SOA Security Manager administrators.

The Connection subelement includes the following parameters:

host

Specifies the host name or IP address of the host where the administrator user store directory is located.

port

Specifies the port number for the administrator user store directory.

Note: The parameters are optional.

User Managed Object Descriptions

You manage user objects that correspond to entries in your administrator user store. Users represent users in your enterprise. A user belongs to a single organization. objects that correspond to entries in a user directory.

An user object description contains the following information:

- Information about the object, such as the LDAP object class and the container in which objects are stored.
- The attributes that store information about an entry. For example, the pager attribute stores a pager number.

Note: A SOA Security Manager environment supports only one type of user object. For example, all user objects have the same object class.

How to Describe a Managed Object

A managed object is described by specifying object information in the User Object section of the directory configuration file.

Each of these sections contains ImsManagedObject elements, such as the following:

```
<ImsManagedObject name="User" description="My Users"
objectclass="top,person,organizationalperson,inetorgperson" objecttype="USER">
```

Specify Object Information

Object information is specified by supplying values for various parameters.

To specify object information

1. Locate the ImsManagedObject element in the User Object section.
2. Supply values for the following parameters:

   Name

   Specifies a unique name for the managed object.

   Note: This parameter is required.

   Description

   Contains a description of the managed object.

   ObjectClass

   Specifies the name of the LDAP object class for the user object. The object class determines the list of available attributes for an object.

   If attributes from multiple object classes apply to an object type, list the object classes in a comma-delimited list. For example, if an object contains attributes from the person, organizationalperson and inetorgperson object classes, add these object classes as follows:

   objectclass="top,person,organizationalperson,inetorgperson"

   Each LDAP directory includes a set of predefined object classes. Refer to the directory server documentation for information about predefined object classes

   Note: This parameter is required.

   ObjectType

   Specifies the type of the managed object. The valid value is User.

   Note: This parameter is required.
3. Optionally, supply container information.

How to Modify Attribute Descriptions

An attribute stores information about an entry, such as a telephone number or address. An entry's attributes determine its profile.

In the directory configuration file, attributes are described in ImsManagedObjectAttr elements. In the User Object section of the directory configuration file, you can do the following:

- Modify default attribute descriptions to describe the attributes in your user store.
- Create new attribute descriptions by copying an existing description and modifying values as needed.

There is one ImsManagedObjectAttr element for each attribute in a user profile. For example, an ImsManagedObjectAttr element may describe a user ID.

An ImsManagedObjectAttr element resembles the following:

```
<ImsManagedObjectAttr physicalname="uid" displayname="User ID" description="User ID" valuetype="String" required="true" multivalued="false" wellknown="%USER_ID%" maxlength="0" />
```

The ImsManagedObjectAttr has the following parameters:

physicalname

This parameter must contain one of the following items:

- The name of the LDAP attribute where the profile value is stored. For example, a user's ID is stored in the uid attribute in the user directory.

  Note: To improve performance, index LDAP attributes that are used in search queries in the Identity Manager User Console.

- A well-known attribute. When you supply a well-known attribute, the Administrative UI computes the value automatically.

description

Contains the description of the attribute

displayname

Specifies a unique name for the attribute.

Note: This parameter is required.

valuetype

Specifies the attribute's data type. The valid values are as follows:

- String (Default value)
- Number
- Integer
- Date

required

Indicates whether the attribute is required, as follows:

- True—The attribute is required
- False—The attribute is optional (default)

Note: If an attribute is required by the LDAP directory server, set the required parameter to true.

multivalued

Indicates whether the attribute can have multiple values. For example, the group membership attribute is multi-valued to store the user DN of each group member. The valid values are as follows:

- True—The attribute can have multiple values
- False—The attribute can have only a single value (default)

Important! The Group Membership and Admin Roles attributes in the User object definition must be multivalued.

wellknown

Defines the name of the well-known attribute.

Well-known attributes have a specific meaning to the Administrative UI. They are identified by the following syntax:

%ATTRIBUTENAME%

maxlength

Defines the maximum length that an attribute's value can have. Set the maxlength parameter to 0 to specify an unlimited length.

Note: This parameter is required.

objectclass

Indicates the LDAP auxiliary class for a user attribute when the attribute is not part of the primary objectclass specified in the ImsManagedObject element.

For example, suppose the primary object class for users is top, person, organizationalperson, which defines the following user attributes:

- common name (cn)
- surname (sn)
- user id (uid)
- password (userPassword)

To include the attribute employeeID, which is defined in the Employee auxiliary class, you would add the following attribute description:

<ImsManagedObjectAttr physicalname="**employeeID**" displayname="**Employee ID**" description="**Employee ID**" valuetype="**String**" required="**true**" multivalued="**false**" maxlength="**0**" objectclass="Employee"/>

How to Describe Attributes

Describing attributes involves the following steps:

1. Read the relevant sections among the following topics:
   - CA Directory Considerations
   - Microsoft Active Directory Considerations
   - IBM Directory Server Considerations
   - Oracle Internet Directory Considerations
2. In the User Object sections of the directory configuration file, do the following:
   - Modify default attribute descriptions to describe your directory attributes.
   - Create new attribute descriptions by copying an existing description and modifying values as needed.

   Note: If you specify a physical attribute when you create a new attribute description, the physical attribute must exist in the object class that you specified for the object type.
3. (Optional) Configure a default sort order.
4. If you are managing a directory with a Flat or Flat User structure or a directory that does not include organizations, go to Describe the User Directory Structure.

CA Directory Considerations

When you describe attributes for a CA Directory administrator user store, note the following:

- Attribute names are case-sensitive.
- Using the photo attribute as the attribute that indicates a user account's status (enabled or disabled) may cause errors when an administrator creates a user.

Note: For additional information about CA Directory requirements, see the CA Directory documentation.

Microsoft Active Directory Considerations

When you describe attributes for Active Directory, note the following:

- The case of the attributes specified in attribute descriptions must match the case of the attributes in Active Directory. For example, when you select the unicodePwd attribute as the attribute that stores user passwords, you must specify unicodePwd (with a capital P) in the directory configuration file.
- For user objects, you must include the sAMAccountName attribute.

Oracle Internet Directory Considerations

When you describe attributes for an Oracle Internet Directory (OID) user store, specify LDAP attributes using lowercase letters only.

Well-Known Attributes for an LDAP User Store

Well-known attributes have special meaning to the Administrative UI. They are identified by the following syntax:

%*ATTRIBUTENAME*%

In this syntax, *ATTRIBUTENAME* must be uppercase.

A well-known attribute is mapped to one physical attribute, using an attribute description.

In the following attribute description, the attribute userpassword is mapped to the well-known attribute %PASSWORD% so that SOA Security Manager will treat the value in userpassword as a password as follows:

```
<ImsManagedObjectAttr
        physicalname="userpassword"
        displayname="Password"
        description="Password"
        valuetype="String"
        required="false"
        multivalued="false"
        wellknown="%PASSWORD%"
        maxlength="0" />
```

Some well-known attributes are required; others are optional.

User Well-Known Attributes

A list of user well-known attributes and the items to which they map follows:

%EMAIL%

Maps to a user's email address.

%ENABLED_STATE%

(Required)

Maps to a user's status.

%FIRST_NAME%

Maps to a user's first name.

%FULL_NAME%

Maps to a user's first and last names.

%LAST_NAME%

Maps to a user's last name.

%PASSWORD%

Maps to a user's password.

%PASSWORD_DATA%

(Required for password policy support)

Specifies the attribute that tracks password policy information.

%USER_ID%

(Required)

Maps to a user's ID.

Configure Well-Known Attributes

Perform the following procedure to configure well-known attributes.

To configure well-known attributes

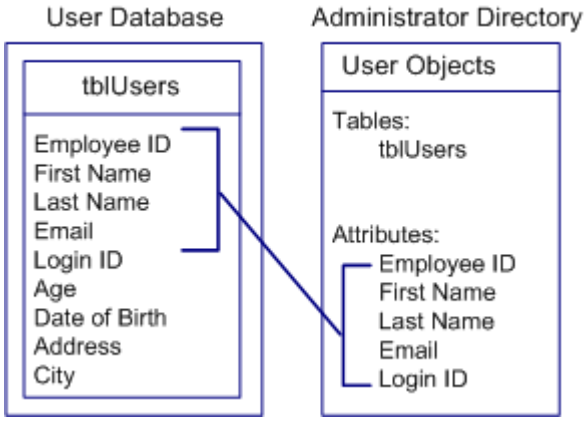
1. In the directory configuration file, search for the following:

   ##
2. Replace the value that begins with ## with the appropriate LDAP attribute.
3. Repeat Steps 1 and 2 until you have replaced all required values.
4. Map optional well-known attributes to physical attributes, as necessary.
5. Save the directory configuration file.

Configure an ODBC Directory Configuration File

A directory configuration file (directory.xml) describes how users are stored in a database. The Administrative UI uses the directory configuration file to locate SOA Security Manager administrators in the database you have designated as your administrator user store.

The following illustration shows how a directory configuration file relates to an administrator user store:



Note: Some user attributes in the database are not part of the directory configuration file, and therefore not required by the Administrative UI.

Before You Configure an ODBC Directory Configuration File

Before you configure an ODBC directory configuration file, ensure that the database meets the following requirements:

- The database must be accessible through an Open Database Connectivity (ODBC) driver. The driver should support outer joins. If more than two tables are used to represent a user object, the driver should also support nested outer joins.

  Note: If the ODBC driver does not support outer joins, the Administrative UI uses inner joins when querying the database. This may cause unexpected query results.

- You must be able to uniquely identify each user object. For example, the unique identifier for users may be a login ID.

  Note: The unique identifier must be stored in a single column.

How to Configure a Directory Configuration File

The steps to configure an ODBC directory configuration file are as follows:

1. Create an ODBC data source for the user for the administrator user store.
2. Describe the database to the Administrative UI by modifying a database-specific directory configuration file (directory.xml).

Create an ODBC Data Source

Define an ODBC data source that points to the administrator user store.

Note: Record the name of the data source. The data source name is required information when you install the Administrative UI.

- For Windows systems, configure the ODBC data source as a System DN. See your Windows operating system documentation for instructions.
- For UNIX systems, add an entry specifying the parameters for the ODBC data source in the system_odbc.ini file. To create the system_odbc.ini file, rename sqlserver.ini to system_odbc.ini. The sqlserver.ini file is located in *policy_server_home*/db.

  policy server home

  Specifies the Policy Server installation path.

How to Describe an Administrator User Store

The Administrative UI must understand the database structure to locate SOA Security Manager administrators. You configure a directory configuration file (directory.xml) to describe the database to the Administrative UI.

SOA Security Manager supplies directory configuration templates that support different database types and structures. To create a directory configuration file, modify the template that most closely matches your directory structure. The templates are located in the DirectoryConfigurationSamples folder at the top level of the Administrative UI installation kit on the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.

3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Copy the configuration template to a new directory or save it with a different name to prevent overwriting it. You can then modify the template to reflect your database structure.

The directory configuration file contains one or more of the following sections:

Directory Information

Contains information about the administrator user store directory that is used by the Administrative UI.

Provider Information

Describes the user store that Administrative UI will manage.

User Object

Describes how users are stored in the administrator user store.

The directory configuration file has two important conventions:

- ##—Indicates required values.

  To provide all of the required information, locate all double pound signs (##) and replace them with appropriate values.

- @—Indicates values that the Administrative UI installer populates. Do not modify these values in the directory configuration file. The Administrative UI installer prompts you to supply the values when you install the Administrative UI.

Before you modify the directory configuration file, you need the following information:

- Table names for the user object
- A list of attributes in user profiles

Modify the Directory Configuration File

Perform the following procedure to modify the directory configuration file.

To modify the directory configuration file

1. Define the user managed objects.
2. Modify well-known attributes.

   Well-known attributes identify special attributes, such as the password attribute, in the administrator user store.

Managed Object Descriptions

You define a user object, which corresponds to entries in the administrator user store. Each entry can be assigned SOA Security Manager administrative privileges.

An object description contains the following:

- Information about the object, such as the tables in which the object is stored.
- The attributes that store information about an entry. For example, the pager attribute stores a pager number.

Important! Only one type of user object is supported.

Describe a Managed Object

A managed object is described by specifying object information in the User Object section of the directory configuration file.

Each of these sections contains an ImsManagedObject element, such as the following:

```
<ImsManagedObject name="User" description="My Users">
```

The ImsManagedObject element may include the following elements:

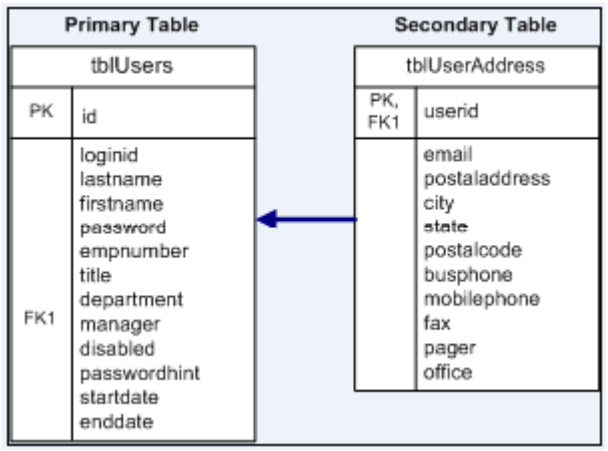
- Table (required)
- UniqueIdentifier (required)
- ImsManagedObjectAttr (required)

Database Tables

Use the Table element in the directory configuration file to define the tables that store information about a managed object.

Each managed object must have one primary table that contains the unique identifier for the object. Additional information may be stored in secondary tables.

The following illustration shows a database that stores user information in a primary and secondary table:



If an object's information is stored in multiple tables, create a Table element for each table. Use the Reference element in the Table element for a secondary table to define its relationship to the primary table.

For example, if basic information about a user is stored in tblUsers and address information is stored in tblUserAddress, the table definitions for the User managed object would resemble the following entries:

```
<Table name="tblUsers" primary="true" />
<Table name="tblUserAddress">
  <Reference childcol="userid" primarycol="id" />
</Table name>
```

Table Elements

The parameters for a table element are as follows:

name

(Required)

Specifies the name of the table that stores some or all of the attributes in a managed object's profile.

primary

Indicates whether the table is the primary table for the managed object. The primary table contains the unique identifier for the object, as follows:

- True—The table is the primary table.
- False—The table is a secondary table (default).

If you do not specify the primary parameter, the Administrative UI assumes that the table is a secondary table.

Note: Only one table can be the primary table.

filter

Identifies a subset of the table entries that apply to the managed object if the table stores information for more than one object type.

The filter parameter may resemble the following:

filter="type='USER'"

Note: The filter applies only to queries that the Administrative UI generates. If you overwrite a generated query with a custom query, you must specify the filter in the custom query.

fullouterjoin

Indicates whether the outer join is a full outer join.

- True— The outer join is a full outer join. In this case, the condition required to return a valid row must be found in both tables in the join for a row to be returned.
- False—The outer join is a left outer join relative to the primary table. In this case, only the rows in one table in the query need to satisfy the condition (default).

Note: The parameters are optional unless otherwise specified.

The Table parameter can contain one or more Reference elements to link a primary table to secondary tables.

Reference Element

The parameters in the Reference element are as follows:

childcol

Indicates the column in the secondary table (specified in the corresponding Table element) that maps to the column in the primary table.

primarycol

Indicates the column in the primary table that maps to the column in the secondary table.

Note: The parameters are optional unless otherwise specified.

Specify Object Information

Object information is specified by supplying values for various parameters.

To specify object information

1. Locate the ImsManagedObject element in the User Object section.
2. Supply values for the following parameters:

   name

   (Required)

   Provides a unique name for the managed object.

   description

   Provides the description of the managed object.

   objecttype

   (Required)

   Specifies the type of managed object. The valid value is USER.

   The ImsManagedObject element should resemble the following:

   ```
   <ImsManagedObject name="User" description="My Users" objecttype="USER">
   ```

3. Supply Table information, as described in Database Tables.
4. Specify the column that contains the unique identifier for the object, as described in Specify the Unique Identifier for a Managed Object.
5. Describe the attributes, as described in Modify Attribute Descriptions.

How to Specify the Unique Identifier for a Managed Object

Each object that the Administrative UI manages must have a unique identifier. The unique identifier must be stored in a single column in the managed object's primary table. Primary tables are described in Database Tables.

Use the UniqueIdentifier and UniqueIdentifierAttr elements to define the unique identifier as follows:

```
<UniqueIdentifier>
 <UniqueIdentifierAttr name="tablename.columnname" />
</UniqueIdentifier>
```

The UniqueIdentifierAttr element requires the name parameter. The value of the name parameter is the attribute in which the unique identifier is stored. The value can be a physical attribute or a well-known attribute.

When you specify a physical attribute, note the following:

- The attribute that you specify must exist in the database and be defined in the directory configuration file, as described in Modify Attribute Descriptions. In the attribute description, be sure to specify read-only or write-once permission to prevent the unique identifier from changing during a session.
- Use the following syntax to specify a physical attribute:

  tablename.columnname

  tablename

  Defines the name of the table where the attribute is located. The table you specify should be the primary table.

  columnname

  Defines the name of the column that stores the attribute.
- If the unique identifier is generated by the database, you must specify a custom operation for the attribute. For example, you may have to specify an operation that fetches the last generated identifier from the database.

Modify Attribute Descriptions

An attribute stores information about a user, such as a telephone number or address. An entity's attributes determine its profile.

In the directory configuration file, attributes are described in ImsManagedObjectAttr elements. In the User Objectsection of the directory configuration file, you can do the following:

- Modify default attribute descriptions to describe your database attributes.
- Create new attribute descriptions by copying an existing description and modifying values as needed.

There is one ImsManagedObjectAttr element for each attribute in userprofiles. For example, an ImsManagedObjectAttr element may describe a user ID.

An ImsManagedObjectAttr element resembles the following:

```
<ImsManagedObjectAttr
 physicalname="tblUsers.id"
 displayname="User Internal ID"
 description="User Internal ID"
 valuetype="Number"
 required="false"
 multivalued="false"
 maxlength="0"
 hidden="false"
 permission="READONLY">
```

Note: When you are using an Oracle database, note the following when you configure managed object attributes:

- Oracle databases are case-sensitive by default. The case of the attributes and table names in the directory configuration file must match the case of the attributes in Oracle.

  Be sure to specify a maximum length for String datatypes to prevent truncation.

The ImsManagedObjectAttr parameters are as follows.

physicalname

(Required)

Specifies the physical name of the attribute, and it must contain one of the following:

- The name and location where the value is stored.

  Format: *tablename.columnname*

  For example, when an attribute is stored in the id column in the tblUsers table, the physical name for that attribute is as follows:

  tblUsers.id

  You must define each table that contains an attribute in a Table element.

  Note: More information on table elements exists in Table elements.

- A well-known attribute.

  A well-known attribute can represent a computed value. For example, you can use a well-known attribute to refer to an attribute that is computed by a custom operation.

displayname

(Required)

Specifies a unique name for the attribute.

description

Provides the description of the attribute.

valuetype

Specifies the attribute's data type. It can be one of the following types:

- String
- Integer
- Number
- Date

When an attribute's valuetype is incorrect, the Administrative UI queries may fail.

required

Indicates whether a value must be specified for the attribute, as follows:

- True—Required
- False—Optional (default)

multi-valued

Indicates whether the attribute can have multiple values, as follows:

- True— An attribute can have multiple values.
- False— An attribute can have only a single value (default).

To store multi-valued attributes in a delimited list instead of in a multi-row table, you must define the delimiter character in the delimiter parameter.

Make sure that the number of possible values and the length of each value that the column enables are sufficient.

wellknown

Provides the name of the well-known attribute.

Well-known attributes have a specific meaning in SOA Security Manager.

Format: %*ATTRIBUTENAME*%

Note: When a custom operation is associated with an attribute, you must specify a well-known attribute.

maxlength

Determines the maximum size of the column.

delimiter

Defines the character that separates values when multiple values are stored in a single column.

Important! The multivalued parameter must be set to true for the delimiter parameter to apply.

Note: The parameters are optional unless otherwise specified.

Custom Operations

You can define custom operations for certain managed objects to do the following:

- Use stored procedures
- Optimize queries for their database structure
- Retrieve a database-generated unique identifier

Custom operations apply only to attributes.

When specifying custom operations, remember the following:

- Users who specify custom operations must be familiar with SQL.
- The Administrative UI does not validate custom operations. Syntax errors and invalid queries are not reported until runtime.
- Custom operations must conform to XML standards. Represent special characters using XML syntax. For example, specify a single quotation mark (') as '

To specify a custom operation, use the Operation element.

Operation Element

The Operation element defines a SQL statement that executes a custom query or calls a stored procedure to create, retrieve, modify, or delete an attribute. It is a subelement of the IMSManagedObjectAttr element, as shown in the following example:

```
<ImsManagedObjectAttr physicalname="tblUsers.id" displayname="User Internal ID" description="User Internal ID" valuetype="Number" required="false" multivalued="false" maxlength="0" hidden="false" permission="READONLY">
  <Operation name="GetDb" value="select @@identity" />
```

Operation element parameters are as follows:

name

Specifies a pre-defined name for an operation. The valid operations are as follows:

- Create
- Get
- Set
- Delete
- GetDB

  The GetDB operation retrieves a unique identifier from the database during a Create task, when the unique identifier is generated by the database or from a stored procedure.

value

Defines the SQL statement or stored procedure to execute. The valid values are as follows:

- INSERT
- SELECT
- UPDATE
- DELETE
- CALL (for stored procedures)

Note: The parameters are optional unless otherwise specified.

The Operation element can contain one or more Parameter elements.

Parameter Element

A Parameter element specifies values that are passed to the query. When multiple Parameter elements are defined, the values are passed to the query in the order in which they are listed.

A Parameter element requires the name parameter. The value of the name parameter can be a physical attribute or a well-known attribute.

Note: The Administrative UI must understand the values that are passed to a query in the Parameter element. For example, the value can be a physical name or a well-known attribute that is defined in the ImsManagedObjectAttr attributes.

When you specify a physical attribute, note the following:

- Use the following syntax to specify a physical attribute:

  tablename.columnname

  - *tablename*

    Provides the name of the table where the attribute is located. The table you specify should be the primary table.

  - *columnname*

    Provides the name of the column that stores the attribute.

- The attribute that you specify must exist in the database and be defined in the directory configuration file, as described in How to Modify Attribute Descriptions.

Example: Custom Operations for the Business Number Attribute

In the following example, the Business Number attribute is generated by calling a stored procedure; it is not a physical attribute in the database.

```
<ImsManagedObjectAttr wellknown="%BUSINESS_NUMBER%"  displayname="Business Number"
description="Business Number"  valuetype="String" required="false" multivalued="false"  maxlength="0">
<Operation name="Get" value="call sp_getbusinessnumber(?)">
  <Parameter name="%USER_ID%"/>
</Operation>
<Operation name="Set" value="call sp_setbusinessnumber(?,?)">
  <Parameter name="%USER_ID%"/>
  <Parameter name="%BUSINESS_NUMBER%"/>
</Operation>
<Operation name="Delete" value="call sp_deletebusinessnumber(?)">
  <Parameter name="%USER_ID%"/>
</Operation>
```

Note the following:

- sp_getbusinessnumber, sp_setbusinessnumber, and sp_deletebusinessnumber are user-defined stored procedures.
- The value returned from the Get operation is mapped to the %BUSINESS_NUMBER% attribute.
- The question mark (?) indicates substitutions that are made at runtime before the query is executed. For example, in the Get operation the %USER_ID% well-known attribute is passed to the sp_getbusinessnumber stored procedure.

Description of a Database Connection

The Provider element and its subelements describe the administrator store database connection.

Note: If you are configuring a directory configuration file for the first time, do not supply connection in the directory.xml file. You provide the connection information for your administrator user store when you install the Administrative UI. The Administrative UI installation wizard prompts you for all of the required database connection information.

Modify the Provider element for updates only.

Provider Element

The Provider element includes the following subelements:

Credentials (required)

Supplies the username and password for accessing the database.

DSN (required)

Identifies the ODBC datasource to use when connecting to the user store.

A completed database connection resembles the following:

```
<Provider type="RDB" userdirectory="@SMDirName">
        <Credentials user="@SMDirUser"
                cleartext="true">@SMDirPassword</Credentials>
        <DSN name="@SMDirDSN" />
```

The parameters for the Provider element are as follows:

type

Specifies the type of database. For Microsoft SQL Server and Oracle databases, specify RDB (default).

userdirectory

Specifies the name of the user directory connection. This parameter corresponds to the name that you supply when installing the Administrative UI.

Database Credentials

To connect to the database, Administrative UI must provide valid credentials to the data source. The credentials are defined in the Credentials element, which resembles the following example:

```
<Credentials user="@SMDirUser" cleartext="true">
 "MyPassword"
</Credentials>
```

Note: Do not supply database credentials in the directory.xml file. You provide the database credentials information for your administrator user store when you install the Administrative UI. The Administrative UI installation wizard prompts you for all of the required credential information.

The credential parameters are as follows:

user

Defines the login ID for an account that can access the data source.

cleartext

Determines whether the password is displayed in clear text in the directory.xml file:

- True—The password is displayed in clear text.
- False—The password is encrypted (default).

Note: These parameters are optional.

Data Source Name (DSN)

The DSN element in the directory.xml file has one parameter: the name of the ODBC data source that the Administrative UI uses to connect to the administrator user store. The value of the name parameter must match the name of an existing data source.

Note: Do not supply the DSN in the directory.xml file. You provide the DSN information for your administrator user store when you install the Administrative UI. The Administrative UI installation wizard prompts you for all of the required DSN information.

Well-Known Attributes for a Relational Database

Well-known attributes have special meaning to the Administrative UI. They are identified by the following syntax:

%*ATTRIBUTENAME*%

In this syntax, *ATTRIBUTENAME* must be uppercase.

A well-known attribute is mapped to one physical attribute using an attribute description.

In the following attribute description, the attribute tblUsers.password is mapped to the well-known attribute %PASSWORD% so that SOA Security Manager will treat the value in tblUsers.password as a password:

```
<ImsManagedObjectAttr
        physicalname="tblUsers.password"
        displayname="Password"
        description="Password"
        valuetype="String"
        required="false"
        multivalued="false"
        wellknown="%PASSWORD%"
        maxlength="0" />
```

Some well-known attributes are required; others are optional.

User Well-Known Attributes

A list of user well-known attributes follows:

%EMAIL%

(Required)

Stores a user's email address

%ENABLED_STATE%

(Required)

Tracks a user's status.

Note: The data type of the physical attribute mapped to %ENABLED_STATE% must be String.

%FIRST_NAME%

Contains a user's first name.

%FULL_NAME%

(Required)

Contains a user's first and last name.

%LAST_NAME%

Contains the user's last name.

%PASSWORD%

Contains a user's password.

%PASSWORD_DATA%

(Required for password policy support)

Specifies the attribute that tracks password policy information.

%USER_ID%

(Required)

Stores a user's login ID.

Configure Well-Known Attributes

Perform the following procedure to configure well-known attributes.

To configure well-known attributes

1. In the directory configuration file, search for the following:

   ##

   Required values are identified by two pound signs (##).

2. Replace the value that begins with ## with the physical name of the attribute you want as it exists in the database. Supply the attribute name using the following format:

   tablename.columnname

   For example, if the password attribute is stored in the password column in the tblUsers table, specify the following:

   tblUsers.password

3. Repeat Steps 1 and 2 until you have replaced all required values and included optional values that you want.
4. Map optional well-known attributes to physical attributes, as necessary.
5. Save the directory configuration file.

How the Administrative UI Installation Works

Installing the Administrative UI is a process that requires access to the machine that is to host the Administrative UI and to the machine that is hosting the Policy Server.

1. **Install the Administrative UI**—The first step in the process is to install the Administrative UI.
2. **Registering the Administrative UI**—The second step in the process is to register the Administrative UI with a Policy. Registering the Administrative UI establishes the connection between the Administrative UI and the Policy Server.

How to Install the Administrative UI

Complete the following procedures to install the Administrative UI:

1. Ensure you have met all of the requirements listed in the Administrative UI pre-installation checklist.
2. Create a database for the object store.
3. Create a database administrator account with privileges to read and write information to the object store.
4. Gather application server information for the Administrative UI installer.
5. Gather object store information for the Administrative UI installer.
6. Gather administrative user store information for the Administrative UI installer.
7. Install the Administrative UI.
8. Start the application server.

Gather Application Server Information

The Administrative UI installer requires specific information about the application server that is installed on the machine that is to host the Administrative UI.

The following sections detail the required information depending on the type of application server in your environment.

Note: Worksheets are provided to help you gather and record information prior to installing and registering the Administrative UI and Report Server. You may want to print these worksheets and use them to record required information prior to installation.

More information:

Administrative UI Installation Worksheets (see page 329)

JBoss Information

If you are using a JBoss application server, gather the following information before installing the Administrative UI:

- **JBoss installation folder—**The path to the folder in which JBoss is installed.

  Note: The path cannot contain spaces.
- **JBoss URL**—The fully qualified URL of the machine on which JBoss is installed.
- **JDK**—The minimum required JDK version for JBoss.

More information:

JBoss Worksheet (see page 329)

WebLogic Information

If you are using a WebLogic application server, gather the following information before installing the Administrative UI:

- **WebLogic binary folder**—The path to the WebLogic installation directory.

  Example: C:\bea\weblogic
- **WebLogic domain folder**—The path to the WebLogic domain you created for the Administrative UI.

  Example: C:\bea\user_projects\domains\mydomain
- **WebLogic server name**—The name of the WebLogic server on which the WebLogic domain is configured.

  Default: AdminServer
- **Application server URL and port**—The fully qualified URL of the machine on which WebLogic is installed.

  Example: http://mymachine.mycompany.com:7001

WebSphere Information

If you are using a WebSphere application server, gather the following information before installing the Administrative UI:

- **WebSphere installation folder**—The full path to the folder in which WebSphere is installed.
- **WebSphere URL**—The fully qualified URL of the machine on which WebSphere is installed.
- **Server name**—The name of the application server.
- **Profile name**—The name of the profile being used for the Administrative UI.
- **Node name**—The name of the node in which the server is located.
- **Cell name**—The name of the cell in which the server is located.
- **JDK**—The minimum required JDK version for WebSphere.

More information:

WebSphere Worksheet (see page 330)

Gather Object Store Information

The Administrative UI installer requires specific information about the database that is to function as the object store.

Gather the following information before installing the Administrative UI.

Note: Worksheets are provided to help you gather and record information prior to installing and registering the Administrative UI and Report Server. You may want to print these worksheets and use them to record required information prior to installation.

- **Host name**—The system name or IP address of the system on which the required JDBC drivers are located.
- **Object store name**—The name of the database instance that you are using as the object store.
- **Object store port number**—The port on which the database instance is listening.
- **Object store administrator account**—The name of the administrator account that has the necessary privileges to read and write information to the database.
- **Object store administrator password**—The password of administrator account that has the necessary privileges to read and write information to the database.

More information:

Object Store Worksheet (see page 330)

Gather Administrative User Store Information

The Administrative UI installer requires specific information about the directory server or database that is storing the Administrative UI administrators. The following sections detail the required LDAP and ODBC information.

Note: Worksheets are provided to help you gather and record information prior to installing and registering the Administrative UI and Report Server. You may want to print these worksheets and use them to record required information prior to installation.

LDAP Administrator User Store Information

If you are using a supported LDAP directory server as an administrator user store, gather the following information before installing the Administrative UI:

- **IP address**—The IP address of the machine that is hosting the directory server.
- **Port number**—The port on which the directory server is listening.
- **Root DN**—The distinguished name of the node in the LDAP tree in which the administrative users are defined.

  Example: dc=security,dc=com
- **Directory manager DN—**The distinguished name of a directory server administrator with the privileges to create, read, modify, and delete objects under the root.

  Example: cn=Directory Manager
- **Directory manager password**—The password for the user with the privileges to create, read, modify, and delete objects under the root.
- **Admin Super User DN**—The distinguished name of a user who is to be the default Super User of the Administrative UI.

  Example:
  uid=superadmin,ou=people,ou=employee,ou=neteauto,dc=security,dc=com
- **Directory.xml file location**—The location of the modified directory.xml file.

  Important! The Administrative UI requires a modified directory.xml to identify users in the administrative user store. If you do not have a modified directory.xml file, do not continue with the installation.

More information:

LDAP Administrative User Store Worksheet (see page 331)

ODBC Administrator User Store Information

If you are using a supported ODBC database as an administrator user store, gather the following information before installing the Administrative UI:

- **Host name**—The name of the machine that is hosting the database.
- **Database port**—The port on which the database is listening.
- **Database name**—The name of the ODBC database you are using as the Administrator user store.
- **Database administrator account**—The user name of an account with privileges to create, read, modify, and delete objects in the database.
- **Database administrator password**—The password for the administrator user account.
- **Admin Super User**—The user name of an account that is to be the default Super User of the Administrative UI.
- **Directory.xml file location**—The location of the modified directory.xml file.

  Note: The Administrative UI requires a modified directory.xml to identify users in the administrative user store. If you do not have a modified directory.xml file, do not continue with the installation.

More information:

ODBC Administrative User Store Worksheet (see page 331)

Install the Administrative UI

The following sections detail how to install the Administrative UI to an existing application server infrastructure.

Run the Windows Installer to Install the Administrative UI

You run the SOA Security Manager installer (ca-soasm-12.1-cr001-win32.exe) to install the Administrative UI. The executable can be downloaded from the Technical Support site.

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.

3. Click Download Center.
4. Search the Download Center for the installation kit you need.

To run the installer to install the Administrative UI

1. Exit all applications that are running.
2. Navigate to where the installer executable is located.

   Important! The installation kit also contains a layout.properties file and a Framework folder at in the Policy Server directory. If you moved the Administrative UI executable after extracting the installation zip, move the following to the same location as the Administrative UI executable or the installation will fail:

   - layout.properties file
   - Framework folder
3. Double-click ca-soasm-12.1-cr001-win32.exe.

   The SOA Security Manager installation wizard starts.
4. Use the gathered application server, object store, and administrative user store information to install the Administrative UI.

   Consider the following when running the installer:

   - When prompted to select features to install, select CA SOA Security Manager UI
   - If you are using IPv6 addresses, ensure your entries include brackets.

     Example: [2001:db8::1428:57ab]
   - When prompted to enter object store information:
     - do not clear the Initialize Database check box.
     - do not enter an IPv6 address. Enter a hostname instead of an IPv6 address. IPv6 addresses are not supported for object store connections.
   - When prompted to enter Tomcat information for the reporting server, ensure that the Tomcat connection port does not conflict with the port number you specified for the application server URL.
5. Review the installation settings and click Install.

   The Administrative UI is installed.

You are now ready to register the Administrative UI with the Policy Server.

Run the UNIX Installer to Install the Administrative UI

The installation executables are in the SOA Security Manager kit available on the Technical Support site. They are the following:

- **Solaris**—ca-soasm-12.1-cr001-sol.bin
- **Red Hat Linux**—ca-soasm-12.1-cr001-linux.bin

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Note: Extract *all* files and directories in the SOA Security Manager kit to a common location on a local drive and keep them together—the SOA Security Manager installer cannot run without the included supporting files.

If you do not have sufficient rights, add executable permission to the install file by running the following command:

chmod+x ca-soasm-12.1-cr001-*os_version*.bin

os_version

Specifies sol or linux.

Important! If you execute the SOA Security Manager installer across different subnets it can crash. To avoid this problem, install the Policy Server directly on the host system.

To run the Administrative UI installer

1. Exit all applications that are running.
2. Open a command window and navigate to where the installation executable is located.
3. Enter the following command in a UNIX shell:

   sh ./ca-soasm-12.1-cr001-*os_version*.bin -i

   The installation starts.

   Note: When prompted to select from a list of numbered choices, enter the numbers separated by commas (,). To select none of the features, enter only a comma.
4. Use the gathered application server, object store, and administrative user store information to install the Administrative UI.

Consider the following when running the installer:

- When prompted to select the features to install, select CA SOA Security Manager UI.
- If you are using IPv6 addresses, ensure your entries include brackets.

  Example: [2001:db8::1428:57ab]
- When prompted to enter object store information:
  - do not clear the Initialize Database check box.
  - do not enter an IPv6 address. Enter a hostname instead of an IPv6 address. IPv6 addresses are not supported for object store connections.
- When prompted to enter Tomcat information for the reporting server, ensure that the Tomcat connection port does not conflict with the port number you specified for the application server URL.

5. Review the installation settings and click Install.

   The Administrative UI is installed.
6. Click Done and reboot the system.

You are now ready to register the Administrative UI with a Policy Server.

Run the Installer from a UNIX Console to Install the Administrative UI

You run the respective UNIX installation executable to install the Administrative UI. The installation executables are in the SOA Security Manager kit available on the Technical Support site. They are the following:

- **Solaris**—ca-soasm-12.1-cr001-sol.bin
- **Red Hat Linux**—ca-soasm-12.1-cr001-linux.bin

To locate installation kits on the Support site

1. Click Technical Support.
2. Log into CA Support Online.

3. Click Download Center.
4. Search the Download Center for the installation kit you need.

Note: Extract *all* files and directories in the SOA Security Manager kit to a common location on a local drive and keep them together—the SOA Security Manager installer cannot run without the included supporting files.

If you do not have sufficient rights, add executable permission to the install file by running the following command:

chmod+x ca-soasm-12.1-cr001-*os_version*.bin

os_version

Specifies sol or linux.

Important! If you execute the SOA Security Manager installer across different subnets it can crash. To avoid this problem, install the Policy Server directly on the host system.

To install the Administrative UI with a UNIX console

1. Exit all applications that are running.
2. Open a command window and navigate to where the install program is located.
3. Enter the following command:

   sh./ca-soasm-12.1-cr001-*os_version*.bin

   The SOA Security Manager installation wizard starts.
4. Use the gathered application server, object store, and administrative user store information to install the Administrative UI.

   Consider the following when entering information:

   - When prompted to select features to install, select CA SOA Security Manager Policy Server
   - When prompted to select components, enter the numbers separated by commas (,). To select none of the features, enter only a comma.
   - If you are using IPv6 addresses, ensure your entries include brackets.

     Example: [2001:db8::1428:57ab]
   - When prompted to enter object store information:
     - do not initialize the database.
     - do not enter an IPv6 address. Enter a hostname instead of an IPv6 address. IPv6 addresses are not supported for object store connections.

5. Review the installation settings and press Enter.

   The Administrative UI is installed.

6. Press Enter.

   The installer closes.

7. Reboot the system.

You are now ready to register the Administrative UI with a Policy Server.

Start the Application Server

Starting the application server deploys the Administrative UI and completes the installation process.

To start the application server

1. Do one of the following:
   - JBoss—From a command prompt, navigate to *jboss_home*\bin.

     *jboss_home**bin*

     Specifies the JBoss installation path.
   - WebLogic—From a command prompt, navigate to *domains*\bin.

     domains

     Specifies the path of the WebLogic domain you created for the Administrative UI.

     Example: C:\bea\user_projects\domains\mydomain
   - WebSphere—From a command prompt, navigate to *profile*\bin.

     profile

     Specifies the path of the WebSphere profile name you created for the Administrative UI.

     Example: C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSvr01\bin
2. Do one of the following:
   - JBoss
     - Windows: Type **run_idm.bat** and press Enter.
     - Solaris: Type **run_idm.sh** and press Enter.
     - Red Hat Linux: Type **run_idm.sh** and press Enter.

- WebLogic
  - Windows: Type **startWebLogic.cmd** and press Enter.
  - Solaris: Type **startWebLogic.sh** and press Enter.
- WebSphere
  - Windows: Type **startServer.bat** *identifier* and press Enter.
  - Solaris: Type **startServer.sh** *identifier* and press Enter.

  identifier

  Specifies the identifier for the WebSphere installation.

  Example: startServer.bat Server1

The application server is started.

Stop the Application Server

To stop the application server

1. Do one of the following:
   - JBoss—From the system hosting the Administrative UI, open the Start Task Engine Command prompt.
   - WebLogic—From the system hosting the Administrative UI, open the Start Task Engine Command prompt.
   - WebSphere—From a command prompt, navigate to *profile*\bin

     profile

     Specifies the path of the WebSphere profile name you created for the Administrative UI

     Example: C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSvr01\bin
2. Do one of the following:
   - JBoss—Enter **Ctrl+c**.
   - WebLogic—Enter **Ctrl+c**.
   - WebSphere
     - Windows: Type **stopServer.bat** *identifier* and press Enter.
     - Solaris: Type **stopServer.sh** *identifier* and press Enter.

     identifier

     Specifies the identifier for the WebSphere installation.

     Example: stopServer.bat Server1

   The application server is stopped.

How to Register the Administrative UI

Registering the Administrative UI requires access to machine that is hosting the Policy Server and the machine that is hosting the Administrative UI. The registration process:

- Establishes a connection between the Administrative UI and the Policy Server.
- (Optional) Creates the initial Super User account.

Complete the following procedures to register the Administrative UI:

1. Run the Administrative UI registration tool.
2. Gather registration information.
3. Configure the Administrative UI and Policy Server connection.

Run the Registration Tool

You run the Administrative UI registration tool to:

- Create a client name and passphrase. A client name and passphrase pairing are values that the Policy Server uses to identify the Administrative UI you are registering. You submit the client and passphrase values from the Administrative UI to complete the registration process.
- (Optional) Create the initial Super User account.

To run the registration tool

1. Open a command prompt from the machine that is hosting the Policy Server.
2. Run the following command:

   XPSRegClient *client_name*[:*passphrase*] -adminui -su -t *timeout* -r *retries*

   -c *comment* -cp -l *log_path* -e *error_path* -vT -vI -vW -vE -vF

   client_name

   Specifies the name that identifies the Administrative UI that is to be registered.

   Limit: This value must be unique. For example, if you have previously used smui1 to register an Administrative UI, enter smui2.

   Note: Record this value. This is a required value to complete the registration process from the Administrative UI.

[:*passphrase*]

Specifies the password required to complete the registration of the Administrative UI.

Limits:

- The passphrase must contain at least 8 characters.
- It must contain at least one uppercase and one lowercase character.
- At least one character must be a digit.
- If the passphrase contains a space, it must be enclosed in quotation marks.
- If you are re-registering the Administrative UI as part of an upgrade, you can reuse a previous passphrase.

Note: If you do not specify the passphrase in this step, xpsregclient prompts you to enter and confirm it.

Important! Record the passphrase, so that you can refer to it later.

-adminui

Specifies that an Administrative UI is being registered.

-su

Specifies that the user that logs into the Administrative UI to complete the registration process becomes the default Super User.

Note: If the Administrative UI that you are registering uses an administrative user store that contains a Super User account established by a previous Administrative UI registration, you do not have to supply this flag. You can use the same Super User account credentials to log into the Administrative UI you are registering to complete the process.

-t *timeout*

(Optional) Specifies how long you have to complete the registration process from the Administrative UI. The Policy Server denies the registration request when the timeout value is reached.

Unit of measurement: minutes

Default: 240 (4 hours)

Minimum Limit: 1

Maximum Limit: 1440 (1 day)

-r *retries*

(Optional) Specifies how may failed attempts are allowed when you complete the registration process from the Administrative UI. A failed attempt is one where you submit an incorrect passphrase to the Policy Server during the registration process.

Default: 1

Maximum Limit: 5

-c *comment*

(Optional) Inserts the specified comments into the registration log file for informational purposes.

Note: Comments must be surrounded by quotes.

-cp

(Optional) Specifies that multiple lines of comments are required. The registration tool prompts for multiple lines of comments and inserts the specified comments into the registration log file for informational purposes.

Note: Comments must be surrounded by quotes.

-l *log path*

(Optional) Specifies the where the registration log file is to be exported.

Default: siteminder_home\log, where siteminder_home is where the Policy Server is installed.

-e *error path*

(Optional) Outputs exceptions to the specified path.

Default: stderr

-vT

(Optional) Sets the verbosity level to TRACE.

-vI

(Optional) Sets the verbosity level to INFO.

-vW

(Optional) Sets the verbosity level to WARNING.

-vE

(Optional) Sets the verbosity level to ERROR.

-vF

(Optional) Sets the verbosity level to FATAL.

The registration tool lists the name of the registration log file and prompts for a passphrase.

3. Press Enter.

   The registration tool creates the client name and passphrase pairing.

You can now register the Administrative UI with the Policy Server. You complete the registration process from the machine hosting the Administrative UI.

Gather Registration Information

The Administrative UI requires specific information about the Policy Server and the client name and passphrase you created to complete the registration process. Gather the following information before logging into the Administrative UI:

Note: Worksheets are provided to help you gather and record information prior to installing and registering the Administrative UI and Report Server. You may want to print these worksheets and use them to record required information prior to installation.

- **Client name**—The client name you specified using the xpsregclient tool.
- **Passphrase**—The passphrase you specified using the xpsregclient tool.
- **Policy Server host**—The IP address or name of the machine hosting the Policy Server
- **Policy Server port**—The port on which the Policy Server is listening.

  Default: 44441

More information:

Administrative UI Registration Worksheet (see page 332)

Configure the Connection to the Policy Server

You configure the Administrative UI and Policy Server connection so SOA Security Manager administrators can use the Administrative UI to manage policy information through the Policy Server. You configure the connection from the Administrative UI.

To configure the Administrative UI and Policy Server connection

1. Open a supported Web browser and enter http://*machine_name*.*company_name*.com:*port*/iam/siteminder.

   The Administrative UI login screen opens.

2. Log in using the credentials of the Super User you identified when installing the Administrative UI.

3. Click Administration, Connections.

4. Click UI, Register Administration UI Server.

   The Register Administration UI Server pane opens.

   Note: Click Help for descriptions of settings and controls, including their respective requirements and limits.

5. Type a connection name in the Name field on the General group box.

6. Type the name or IP address of the machine on which the Policy Server is installed in the Policy Server Host field.

7. Type the port on which the Policy Server is listening in the Policy Server Port field.

   Note: This value must match the value in the Accounting port (TCP) field on the Settings tab in the Policy Server Management Console. The default accounting port is is 44441. To determine a non-default port number, open the Settings tab in the Policy Server Management Console.

8. Type the client name and passphrase you created using the registration tool in the fields on the General group box.

9. Select a FIPS compatibility mode:

   - If you installed the Policy Server in FIPS-compatibility mode, select Compatibility mode.
   - If you installed the Policy Server in FIPS-only mode, select FIPS only mode.

10. Click Submit.

    The connection between the Administrative UI and Policy Server is configured. The Infrastructure, Policies, and Reports tab are now available.

    Note: The Administrative UI login screen contains a list of Policy Servers to which the Administrative UI is registered. By default, the Policy Server that was registered first appears at the top of the list. More information about changing which Policy Server connection appears at the top of the list exists in Modify the Default Policy Server Connection.

You are now ready to prepare for the Web Agent installation.

Prepare for Web Agent Installation

Before you install a Web Agent, you must have:

- Installed the Policy Server.
- Configured a policy/key store to communicate with the Policy Server.
- Installed and registered the Administrative UI.
- Confirmed that the Policy Server can communicate with the system on which you will install the Web Agent.

Before you can register a trusted host at the Web Agent site, the following objects must be configured in the Administrative UI.

Note: For more information about configuring each of the following objects, see the *Policy Server Configuration Guide*.

To centrally manage Agents, configure the following using the Administrative UI:

- **A SOA Security Manager Administrator that has the right to register trusted hosts**—A trusted host is a client computer where one or more SOA Security Manager Web Agents are installed. The term trusted host refers to the physical system. There must be an administrator with the permission to register trusted hosts. The default SOA Security Manager administrator has this permission.
- **Agent identity—**An Agent identity establishes a mapping between the name and the IP address of the web server instance hosting a Web Agent. You define an Agent identity from the Agents object in the Administrative UI. You assign the Agent identity a name and specify the Agent type as a Web Agent.

  Note: The name you assign for the Agent is the same name you specify in the DefaultAgentName parameter for the Agent Configuration Object.

- **Host Configuration Object—**A host configuration object defines the communication between the trusted host and the Policy Server after the initial connection between the two is made.

  Do not confuse the host configuration object with the trusted host configuration file, SmHost.conf, which is installed at the trusted host after a successful host registration. The settings in the SmHost.conf file let the host connect to a Policy Server for the first connection only. Subsequent connections are governed by the host configuration object.

- **Agent Configuration Object—**An Agent configuration object includes the parameters that define the Web Agent configuration. There are a few required parameters you are required to set for the basic operation described below.

  Note: For more information about Agent parameters, see the *Web Agent Configuration Guide*.

  - **For all Agents—**The Agent Configuration Object must include a value for the DefaultAgentName. The DefaultAgentName must match the Agent identity name you specified in the Agents object. The DefaultAgentName identifies the Agent identity that the Web Agent uses when it detects an IP address on its web server that does not have an Agent identity assigned to it.

  - **For Domino Web Agents—**The Agent Configuration Object must include values for the following parameters:

    - **DominoDefaultUser—**If the user is not in the Domino Directory, and they have been authenticated by SOA Security Manager against another user directory, this is the name by which the Domino web agent identifies that user to the Domino server. The DominoDefaultUser value can be encrypted.

    - **DominoSuperUser—**Ensures that all users successfully logged into SOA Security Manager are logged into Domino as the DominoSuperUser. The DominoSuperUser value can be encrypted.

  - **For IIS Web Agents—**The Agent Configuration Object must include values for the DefaultUserName and DefaultPassword parameters. The DefaultUserName and DefaultPassword identify an existing Windows account that has sufficient privileges to access resources on an IIS web server protected by SOA Security Manager. When users need to access resources on an IIS web server protected by SOA Security Manager, they may not have the necessary server access privileges. The Web Agent must use the Windows account, which is previously assigned by an administrator, to act as a proxy user account for users granted access by SOA Security Manager.

  Note: If you plan to use the NTLM authentication scheme, or enable the Windows User Security Context feature, do not specify values for these IIS Web Agent parameters.

Modify the Default Policy Server Connection

The Administrative UI login screen contains a list of Policy Servers to which the Administrative UI is registered. By default, the Policy Server that was registered first appears as the default connection. You can modify the list to have another Policy Server connection appear as the default.

To modify the default Policy Server connection

1. Click Administration, Admin UI.
2. Click Policy Server Connections, Modify Policy Server Connection.

   The Modify Policy Server Connection pane appears.
3. Specify search criteria and click Search.

   Administrative UI connections matching the criteria appear.
4. Select the connection you want and click Select.

   Settings specific to the Administrative UI connection appear.
5. Click the arrow icon in the Advanced group box.

   Advanced settings appear.
6. Select the Default Connection check box and click Submit.

   The Policy Server connection is configured as the default connection.

Uninstall the Administrative UI on Windows

You uninstall the Administrative UI when it is no longer required on the system.

To uninstall the Administrative UI

1. Stop the application server.
2. Open the Windows Control Panel and double-click Add/Remove Programs.

   A list of installed programs appears.
3. Select CA SOA Security Manager Web Access Manager Administrative UI, and click Change/Remove.

   The process to uninstall the Administrative UI starts.
4. Follow the instructions and prompts in the wizard.

   Note: If a message prompts you to remove shared files, click No to All.
5. If requested, reboot the system.

   The Administrative UI is uninstalled.

More information:

Installing the Administrative UI (see page 243)

Uninstall the Administrative UI on UNIX

You uninstall the Administrative UI when it is no longer required on the system.

To uninstall the Administrative UI on UNIX

1. Stop the application server.
2. Navigate to the following directory in a console window:

   smwamui_home/CA/smwamui/install_config_info/sm-wamui-uninstall

   smwamui_home

   Specifies the Administrative UI installation path.
3. Run the following command:

   ./uninstall
4. Press Enter.

   A status indicator shows progress and prompts you when completed.
5. If you installed the Administrative UI as a:
   - Root user, navigate to the /var directory.
   - Non-root user, navigate to user_home.
6. Delete the .CA_IAM_FW.registry file.
7. Open the file .com.zerog.registry.xml file.
8. Delete only the section that begins <feature name="Framework"... and ends </feature>, and save the file.

   The Administrative UI is removed from the system.

More information:

Installing the Administrative UI (see page 243)

Chapter 9: Registering the Federation Security Services Administrative UI

This section contains the following topics:

Registering the FSS Administrative UI

The FSS Administrative UI is an applet-based application that is installed with the Policy Server and is used to manage eTrust SiteMinder FSS. eTrust SiteMinder FSS components consist of the affiliates (consumers, service providers, resource partners) and SAML authentication schemes that you configure to support federated communication between two partners.

The intent of the FSS Administrative UI is to let you manage SOA Security Manager eTrust SiteMinder FSS. If you are familiar with previous versions of the SiteMinder Policy Server User Interface, you will notice that all SiteMinder objects appear in the FSS Administrative UI. The only objects that do not appear are objects related to Enterprise Policy Management (EPM) and reports. You can use the FSS Administrative UI to manage the SOA Security Manager objects. If you need information while using the FSS Administrative UI, consult the FSS Administrative UI online help system.

You register the FSS Administrative UI with the Policy Server to ensure that the communication between both components is FIPS-encrypted (AES encryption).

Note: If you organization is not federating with a partner, you may safely leave the FSS Administrative UI on the Policy Server machine without registering it with the Policy Server.

Installation Road Map

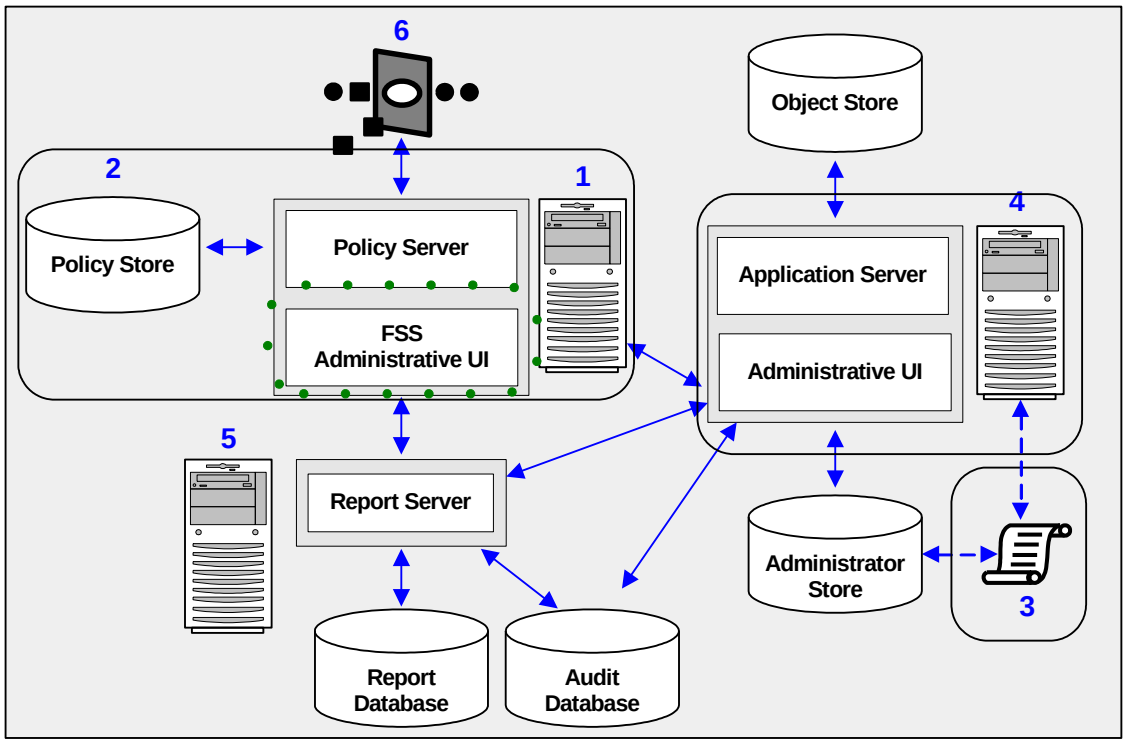
The following diagram illustrates a sample SOA Security Manager installation and lists the order in which you install and configure each component.

- The components surrounded by a solid line represent the Policy Server, policy store, and a non-registered FSS Administrative UI; a configured directory XML file; and an installed and registered Administrative UI.

  Note: The FSS Administrative UI is installed with the Policy Server.

- The component surrounded with a green dotted line represents the FSS Administrative UI, which you register with the Policy Server at this point in the installation process.

  Note: A registered FSS Administrative UI is only required to manage eTrust SiteMinder FSS. If you organization is not federating with a partner, you may safely leave the eTrust SiteMinder FSS on the Policy Server machine without registering it with the Policy Server.



The following sections detail:

- The pre-requisites for registering the FSS Administrative UI
- How to register the FSS Administrative UI

Pre-registration Checklist

You may want to print the following to use as a checklist to help ensure that you have the required SOA Security Manager components and information to register the FSS Administrative UI with the Policy Server:

- ☐ Ensure that the Administrative UI is installed and configured. You register the FSS Administrative UI using the Administrative UI.
- ☐ Ensure that the Administrative user account you will use to access the Administrative UI has the necessary privileges to create an agent. You create a 4.x Agent during the FSS Administrative UI registration process.
- ☐ Ensure that you have the default SOA Security Manager Super User account password. You log in to the FSS Administrative UI to confirm the registration process.

  Note: The default SOA Security Manager Super User account is SOA Security Manager. The default SOA Security Manager Super User account password was created when the policy store was configured. If you do not have the password, you may change it using the smreg utility. More information on using the smreg utility exists in the *Policy Server Administration Guide*.
- ☐ If you are registering the FSS Administrative UI as part of an upgrade to r12.1, ensure that you have upgraded the policy store to r12.1. You should not begin using the FSS Administrative UI to manage eTrust SiteMinder FSS until the policy store is upgraded to r12.1.

Before You Register the FSS Administrative UI

Do the following before you register the FSS Administrative UI:

- (Internet Explorer) If you are using a supported version of Internet Explorer, add your domain as a trusted site before accessing the FSS Administrative UI.

  Consider the following when adding your domain as a trusted site:

  - Enter the full name of the server, including the domain.

    Example: http://*servername.domain-name*

    Example: http://security.myorg.org

  - If you are connecting to the FSS Administrative UI using a secured connection (https), you must include https when specifying the domain.

    Example: https://security.myorg.org

  - If you are not accessing the Policy Server using a secured connection (https), you may disable the Require server verification (https) for all sites in this zone setting.

  Note: For more information about adding a domain as a trusted site, see the Internet Explorer documentation.

- (Netscape Communicator) If you are using a supported version Netscape Communicator, you can reduce the time the FSS Administrative UI takes to load by copying FSS Administrative UI sm_admin_noswing.jar file to the directory where the Netscape Java classes are located. The sm_admin_noswing.jar file is located in *policy_server_home*\admin.

  policy_server_home

  Specifies the Policy Server installation location.

  Note: For more information about where Netscape Java classes are located, see your Netscape documentation.

- (Windows) When launching the FSS Administrative UI, the URL assumes that you are running the Web server at the default port 80. If the Web server is not running on the default port, or if you change the Web server port, modify the FSS Administrative UI shortcut in the Start menu before accessing the FSS Administrative UI.

- Accessing the FSS Administrative UI locally reduces the time it takes to load. If you access the FSS Administrative UI from Internet Explorer, the FSS Administrative UI files, which are stored in sm_admin.cab, are automatically stored in Internet Explorer's cache.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site.

How to Register the FSS Administrative UI

Complete the following steps to register the FSS Administrative UI with the Policy Server:

1. Review pre-requisite considerations in Before you Register the FSS Administrative UI.
2. Create the registration credentials for the FSS Administrative UI.
3. Log into the FSS Administrative UI.

Create the Registration Credentials for the FSS Administrative UI

In addition to the SOA Security Manager Super User administrator account credentials, which were established when you installed the Policy Server and configured the policy store, the FSS Administrative UI login screen also requires the name and shared secret of a 4.x Agent. You establish these credentials to ensure that the communication between the FSS Administrative UI and the Policy Server is FIPS-encrypted (AES encryption).

Note: Creating a 4.x Agent identity to log into the FSS Administrative UI does not require that you to install a SOA Security Manager Web Agent, configure a Host Configuration Object and Agent Configuration object, or register a trusted host. The FSS Administrative UI only requires the 4.x Agent identity.

To create the 4x agent

1. Log into the Administrative UI.
2. Click Infrastructure, Agents.
3. Click Agent, Create Agent.

   The Create Agent screen appears. The create new object radio button is selected.
4. Click OK.

   Agent-related properties appear.
5. Enter an Agent name in the Agent Name field.

   Example: FSS UI Agent

   Note: Record the Agent name as it is required to log into the FSS Administrative UI.
6. Select the Supports 4.x agents check box in the Agent Type Settings group box.

   The Trust Settings group box appears.

7. Enter the IP Address of the machine hosting the Policy Server in the IP Address field.
8. Enter and confirm the Agent shared secret in the Shared Secret and Confirm Shared Secret fields.

   Note: Record the Agent shared secret as it is required to log into the FSS Administrative UI.

   You have created the required 4.x Agent credentials and may log into the FSS Administrative UI.

Log into the FSS Administrative UI

You log into the FSS Administrative UI to complete the registration process.

To log into the FSS Administrative UI

1. On the Status tab of the Policy Server Management Console, ensure that the Policy Server is running.

   Note: More information on accessing the Policy Server Management Console exists in the *Policy Server Administration Guide*.

2. Complete one of the following

   – Enter http://*hostname*:*port*/siteminder in a supported browser.

   hostname:port

   Specifies the name of the machine on which the Policy Server is installed. The port is the port number of the Web server. You do not need to enter a port number if you are using the default port (80) for HTTP requests.

   Example: http://www.myorg.org:80/siteminder

   – From the Start menu, select Start, Programs, siteminder, SOA Security Manager FSS Administrative UI

   The FSS Administrative UI login screen appears.

   Note: If the login screen does not appear, refer to Federation Security Services Administrative UI Troubleshooting.

3. Click Login.

   Note: If you are prompted to trust or grant permissions to the signed applet delivered by CA, click Yes.

   The SOA Security Manager FSS Administrative UI login screen appears.

4. Enter the following information:
   - SiteMinder in the Username field
   - The password for the SOA Security Manager Super User account in the Password field.
   - The name of the 4.x Agent identity you created in the Host Name field.
   - The shared secret of the 4.x Agent identity you created in the Passphrase field.
5. Click Login.

   The FSS Administrative UI opens.

Note: We recommend that the default SOA Security Manager Super User account not be used in day-to-day operations. Instead, create an administrator with system configuration privileges. More information on creating a FSS Administrative UI administrator exists in the *FSS Administrative UI Help* system.

Chapter 10: Configuring the OneView Monitor

This section contains the following topics:

OneView Monitor Overview

The OneView Monitoring infrastructure consists of a number of modules that enable the monitoring of SOA Security Manager components. Included is the Monitor process that runs in the context of a Java Runtime Environment (JRE). The Monitor GUI's HTML pages are generated by Java Server Pages (JSPs) and servlets hosted on a ServletExec servlet engine running on the same machine as the Policy Server.

The OneView Monitor utility monitors the following SOA Security Manager components:

- Policy Server
- Web Agents

Note: More information on using the OneView Monitor exists in the *Policy Server Administration Guide*.

System Requirements for OneView Monitor

The system to which you are configuring the OneView Monitor GUI must meet at least the following system requirements:

- **JDK**—The required version of the Java SDK is installed on the system.
- **Servlet Engine**—The required ServletExec/ISAPI for Windows or ServletExec/AS for UNIX is installed on the system.
- **Web server**—A supported Web server is installed on the system.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site.

Configure the OneView Monitor

If you did not configure the OneView Monitor GUI when installing the Policy Server, you can configure it using the Policy Server Configuration Wizard.

You can find the following Policy Server Configuration Wizard executables in *siteminder_home*\siteminder\install_config_info for Windows and *siteminder_home*/siteminder/install_config_info for UNIX:

- ca-ps-config.exe
- ca-ps-config.bin

siteminder_home

Specifies the path to where the Policy Server is installed.

Limitation of OneView Monitor GUI/IIS Web Agent on Same Machine

CA does not support the configuration of the IIS-based OneView Monitor GUI and the IIS Web Agent on the same machine if the Agent has Registration Services enabled. With this configuration, there is a conflict with the same instance of ServletExec.

How to Configure the OneView Monitor GUI on Windows/IIS

To configure the OneView Monitor GUI on Windows/IIS complete the following procedures:

1. Read the prerequisites to installing ServletExec on Windows.
2. Install ServletExec/ISAPI on Windows/IIS.

   Note: The Monitor GUI's HTML pages are generated by Java Server Pages (JSPs) and servlets hosted on a ServletExec servlet engine running on the same machine as the Policy Server.
3. Assign modify permissions to the Internet guest account for the *policy_server_home*\monitor\settings folder.
4. Set permissions for the IIS Users.
5. If you did not have the Policy Server installation program auto-configure the OneView Monitor GUI, configure it by running the Policy Server Configuration Wizard.
6. Start the OneView Monitor service.
7. Access the OneView Monitor GUI.

Prerequisites to Installing ServletExec on Windows

CA recommends that you read the ServletExec documentation before installing ServletExec. If ServletExec is not running properly, then the OneView Monitor GUI does not work since it relies on ServletExec's servlet engine.

Note: You can access the ServletExec documentation on the New Atlanta Web site.

Install ServletExec/ISAPI on Windows 2003/IIS

To install ServletExec/ISAPI on Windows 2003/IIS

1. If you have an earlier version of ServletExec:
   a. Back up the ServletExec Data and Servlets sub-directories, if desired.
   b. Remove the earlier version.
2. Run the ServletExec ISAPI installer.

   Note: For more information on running the ServletExec ISAPI installer, refer to New Atlanta Communication's ServletExec documentation.
3. Stop and restart the IIS Admin Web service and IIS Web server.

Set Permissions for IIS Users After Installing ServletExec

Since ServletExec/ISAPI runs as part of the IIS process, it runs as different users at different times. As a result, you must set the following permissions for the ServletExec installation directory and subdirectories.

To set permissions for IIS users after installing ServletExec, Make sure the user that runs IIS (for example, Network Services) has read and write access to the entire directory tree under C:\Program Files\New Atlanta.

How to Configure the OneView Monitor GUI on UNIX/Sun Java System

To configure the OneView Monitor GUI on a UNIX/Sun Java System complete the following procedures:

1. Read the prerequisites to installing ServletExec.
2. Disable servlets in Sun Java System (Sun One/iPlanet) 6.0.
3. Install ServletExec/AS on UNIX/Sun Java System.
4. If you did not have the Policy Server installation program auto-configure the OneView Monitor GUI, configure it by running the Policy Server Configuration Wizard.
5. Start the OneView Monitor Service.
6. Access the OneView Monitor GUI.

Prerequisites to Installing ServletExec

CA recommends that you read the ServletExec documentation before installing ServletExec. If ServletExec is not running properly, then the OneView Monitor GUI does not work since it relies on ServletExec's servlet engine.

You can access the ServletExec documentation on the New Atlanta Web site.

Disable Servlets in Sun Java System 6.0

Ensure you follow the steps in this section before installing ServletExec.

To disable servlets in Sun Java System 6.0

1. Open the Sun Java System Enterprise Administration Server home page by entering the following URL in a browser:
   http://<yourserver.com>:<portnumber>

   yourserver.com

   Specifies the domain name of the Enterprise Administration Server

   port

   Specifies the port number

2. In the Select a Server drop-down menu, select the target server, and then click Manage.
3. Select the Java tab.
4. Deselect Enable Java for class defaultclass and Enable Java Globally and click OK.
5. Stop and restart the Web server so the settings can take effect.

Install ServletExec/AS on UNIX/Sun Java System

The Monitor GUI's HTML pages are generated by Java Server Pages (JSPs) and servlets hosted on a ServletExec servlet engine running on the same machine as the Policy Server.

To install ServletExec

1. Log in to the UNIX account where you want to install the Policy Server.

   Note: You must log in as the same user who installed the Sun Java System Web server.

2. Run the ServletExec AS installer.

   Note: For more information on running the ServletExec AS installer, refer to New Atlanta Communications' ServletExec documentation. Consider the following before installing ServletExec:

   - Make sure you have permission to create a new file in /tmp. New Atlanta recommends installing ServletExec in /usr/local/NewAtlanta. Installing ServletExec in /usr/local/NewAtlanta may change the permissions for the obj.conf file and the Sun Java System start script. After the installation, be sure the owner of obj.conf and the start script is the same user who owns the Web server.
   - When prompted, install a Web server adaptor and an instance of ServletExec.
   - When prompted, ensure that the installer does not modify the Web server's configuration files. If you let the installer modify the Web server's obj.conf and magnus.conf configuration files, the Web server instance fails to run after you configure the OneView Monitor GUI on this instance.

3. After the installation program completes, restart the Web server.

More Information:

Start the OneView Monitor Service (see page 313)

Start the OneView Monitor Service

To start the OneView Monitor service

1. Make sure the IPC port numbers are available.

   The OneView Monitor uses the following port numbers to communicate with the Policy Server processes:

   - Monitoring Agent: 44449
   - Monitor: 44450

   To see which port numbers are unavailable, open a Command Window and enter:

   ```
   netstat -an
   ```

   Note: For more information on changing the port numbers, see the Policy Server Administration Guide.

2. Using the Status tab of the Policy Server Management Console, start the Monitor service.

Access the OneView Monitor GUI

To access the OneView Monitor GUI

Enter the following URL in a browser:

http://*server:*<*portnumber*>/siteminder**monitor**

server

Specifies the Web Server's IP Address

portnumber

Specifies the port number.

Monitor a Policy Server Cluster

The OneView Monitor can be configured to monitor a Policy Server cluster when one Policy Server is set up as a centralized monitor for other Policy Servers in a cluster.

Note: More information on using the OneView Monitor exists in the *Policy Server Administration Guide*.

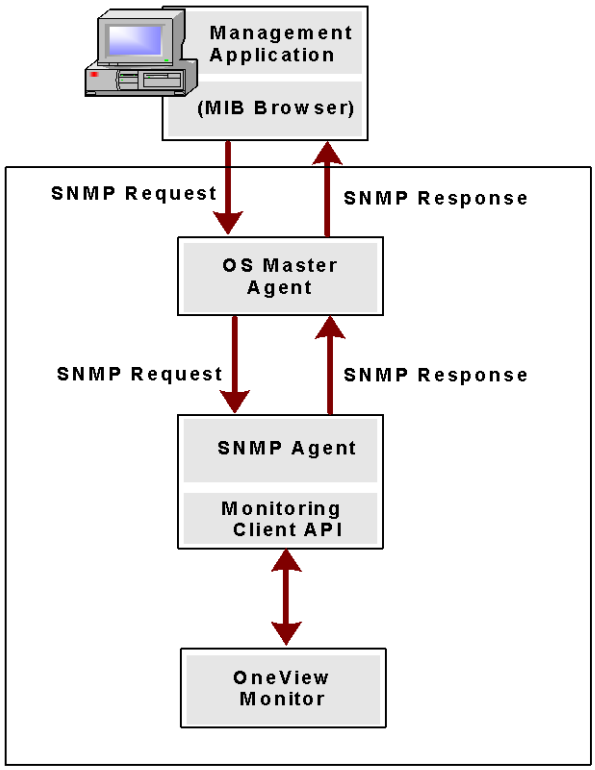
Chapter 11: SNMP Support

This section contains the following topics:

SNMP Support Overview

SNMP support includes a Management Information Base (MIB), an SNMP Agent, and the Event SNMP Trap library. You can configure the SNMP Agent and Event SNMP Trap library independently and enable one or disable the other or vice versa. The SNMP Agent enables monitoring applications to retrieve operational data from the OneView Monitor. The SNMP Agent sends data to the SNMP manager and supports SNMP request handling.

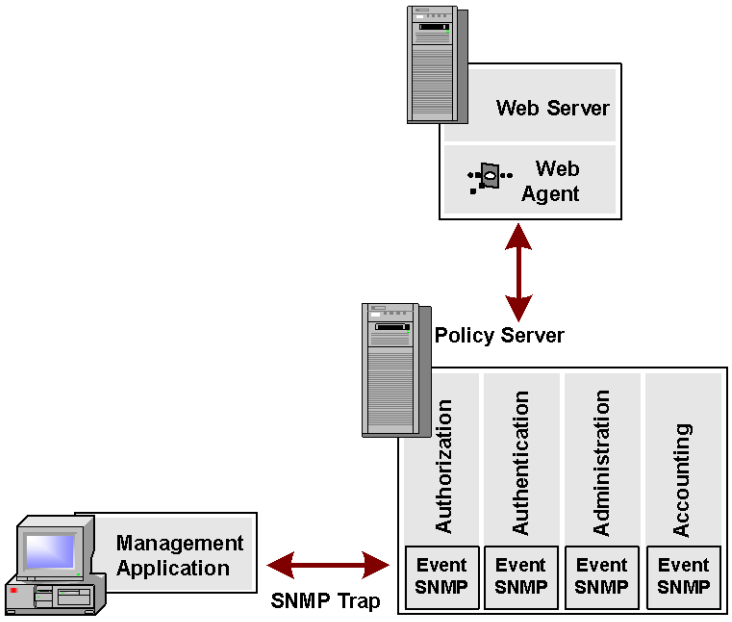
The following figure shows the architecture between the management application, OS Master Agent, SNMP Agent, and the OneView Monitor.



The OS Master Agent, such as the native Solaris SunSolstice Master Agent, invokes the SNMP Agent once you restart the Master Agent. Upon receiving an SNMP request from the management application the OS Master Agent forwards the SNMP request to the SNMP Agent. The SNMP Agent contacts the OneView Monitor, retrieves the required information using Monitor Client API, and then sends the response to the Master Agent. The Master Agent, in turn, forwards the response to the management application.

If you do not configure the SNMP Agent during the Policy Server installation, all the SNMP files are still installed in case you want to use the Agent later. However, to get the Agent running, you need to manually get the Agent started by configuring the SNMP Agent on a Windows or UNIX system.

The Event SNMP Trap library converts some SOA Security Manager events into SNMP traps before sending them to the management application as noted in the following figure. The trap library captures events sent by the Policy Server, decides if SNMP traps are to be generated on a given event, and generates a trap.



Note: For more information on the SNMP Agent and the OneView Monitor, see the *Policy Server Administration Guide*.

Prerequisites for Windows and UNIX Systems

You need to have a Master Agent installed with your operating system before installing or using the SNMP Agent.

Windows Prerequisites

SOA Security Manager SNMP support on Windows requires the SNMP service. For more information about installing the SNMP service, see the Windows online help system.

UNIX Systems Prerequisites

The following section details UNIX prerequisites for SNMP support:

Solaris

You need the native Solaris SunSolstice Master Agent, which comes with the operating system.

Linux

For the supported Master Agent on Red Hat Advanced Server 3.0, upgrade the net-snmp package to net-snmp-5.1-2.1 or greater.

To upgrade the net-snmp package to net-snmp-5.1-2.1 or greater, use the following setting in net-snmpd instead of-c public -v 1 -p 8001 localhost .1.3.6.1.4.1.2552:

proxy -c public -v 1 localhost:8001 .1.3.6.1.4.1.2552

Configure the SNMP Agent on Windows

To configure the SNMP agent on Windows

1. Be sure that the NETE_PS_ROOT environment variable is set to the SOA Security Manager installation directory. The Policy Server installation program should have already done this.
2. Open *siteminder_home*\config\snmp.conf file and edit the last row to contain the full path to *siteminder_home*\log\snmp.log.

   Note: You only need to do this if you did not specify the Policy Server installation program to automatically configure SNMP.

   Correct example: LOG_FILE=C:\Program Files\Netegrity\siteminder\log\snmp.LOG

   Incorrect example: LOG_FILE=$NETE_PS_ROOT\log\snmp.log
3. Edit the w*indows_dir*/java_service.ini file.

   Note: You only need to do this if you did not specify the Policy Server installation to automatically configure SNMP.

   a. Set SERVICE_BINARY_NAME to the full path name of JavaService.exe.

      Example: SERVICE_BINARY_NAME=c:\winnt\JavaService.exe

   b. Set WORKING_DIR to the full path to directory *siteminder_home*\bin:

      Example: WORKING_DIR=C:\Program files\Netegrity\siteminder\bin

   c. Set JRE_PATH to the full path of javaw.exe.

4. Run *siteminder_home*\bin\thirdparty\proxyreg.exe to change the registry keys for the apadll.dll and snmp.conf:

   proxyreg.exe *full_path_for_apadll.dll full_path_for_snmp.conf*

   Important! If you are running a SOA Security Manager utility or executable on Windows Server 2008, be sure to open the command–line window with Administrator permissions, even if you are logged into the system as an Administrator. For more information, see the release notes for your SOA Security Manager component.

   Example: proxyreg.exe "c:\program files\netegrity\siteminder\ bin\ thirdparty\apadll.dll" "c:\programfiles\netegrity\ siteminder\ config\ snmp.conf"

5. Run *WINNT dir*/JavaService.exe with the -install option, to register the Netegrity SNMP agent as a WINNT service.
6. Start the Netegrity SNMP agent by using the Windows Services dialog box.
7. Restart the SNMP service.

How to Configure SNMP Event Trapping on Windows

Configuring SNMP event trapping on Windows requires you to:

1. Enable SNMP event trapping.
2. Configure snmptrap.config (see page 320).

Enable SNMP Event Trapping

To enable SNMP event trapping, use the XPSConfig utility to set the event handler library (eventsnmp.dll) to the XPSAudit list. The default location of eventsnmp.dll is *policy_server_home*\bin.

policy_server_home

Specifies the Policy Server installation location.

Note: More information on using the XPSConfig utility to set event handler libraries exists in the *Policy Server Administration Guide*.

After enabling SNMP event trapping, configure the snmptrap.conf file.

Configure snmptrap.conf

To configure the SNMP configuration file

1. Edit snmptrap.conf.

   Note: snmptrap.conf is located in *policy_server_home*\config.

   policy_server_home

   Specifies the Policy Server installation location.

2. For the specified trap(s) that you want to receive, uncomment out the appropriate line(s).
3. Specify the IP Address, port number, and community for where you want the trap to be sent.
4. Save the snmptrap.config file with the new changes.
5. Restart the Policy Server.

More Information:

Stop and Restart the Policy Server (see page 322)

Configure the SNMP Agent on UNIX Systems

To configure the SNMP Agent on UNIX systems

1. Ensure the NETE_PS_ROOT environment variable is set to the SOA Security Manager installation directory. The Policy Server installation program should have already done this.

   Example: /home/smuser/siteminder

2. Edit the file /etc/snmp/conf/RunSubagent.sh:

   a. Set the correct JRE path:
   JAVA_HOME=$INSTALL_HOME/bin/jdk/<required_version>/jre

   b. Set the correct SOA Security Manager path:

   Example: INSTALL_HOME=/home/smuser/siteminder

   Note: The INSTALL_HOME variable should contain the full path for the SOA Security Manager installation directory.

3. Restart the SNMP daemon on Solaris
   a. Become root.
   b. Goto /etc/rc3.d.
   c. Execute the S76snmpdx script twice, as follows:

      sh S76snmpdx stop to stop the running Solaris master agent.

      sh S76snmpdx start to start the Solaris master agent and Netegrity subagent.

How to Configure SNMP Event Trapping on UNIX Systems

Configuring SNMP event trapping on UNIX systems requires you to:

1. Enable SNMP event trapping.
2. Configure snmptrap.config (see page 321).

Enable SNMP event trapping

To enable SNMP event trapping, use the XPSConfig utility to set the event handler library (libeventsnmp.so) to the XPSAudit list. The default location of libeventsnmp.so is *policy_server_home*/lib.

policy_server_home

Specifies the Policy Server installation location.

Note: More information on using the XPSConfig utility to set event handler libraries exists in the *Policy Server Administration Guide*.

After enabling SNMP event trapping, configure the snmptrap.config file.

Configure snmptrap.config

To configure snmptrap.config

1. Edit snmptrap.config, which is located in /home/smuser/siteminder/config.
2. For the specified trap(s) that you want to receive, uncomment out the appropriate line(s).
3. Specify the IP Address, port number, and community for where you want the trap to be sent.
4. Save the snmptrap.config file with the new changes.
5. Restart the Policy Server.

More Information:

Stop and Restart the Policy Server (see page 322)

Stop and Restart the Policy Server

In order for the SNMP configurations changes to take effect, you need to stop and restart the Policy Server using the Status tab of the Policy Server Management Console.

Test SNMP Gets for Red Hat Enterprise Linux Advanced Server

You should test SNMP Gets after configuring SNMP.

To test SNMP Gets

1. Start the native SNMP master agent. On Red Hat AS, the master agent is not started automatically on start up as is the case on Solaris and HP-UX. To start the master agent, go to the /etc/rc1.d directory and run the following command (run as root):

   K50snmpd start

2. Start the Netegrity subagent using the following command (run as root):

   sh /etc/init.d/NetegrityAgent

3. To stop the Netegrity subagent on Red Hat AS, run the following command as root:

   sh $NETE_PS_ROOT/etc/snmp/conf/StopSubagent.sh

Test SNMP Gets for HP-UX

You should test SNMP Gets after configuring SNMP.

To test SNMP Gets

1. Start the Native Agent Adaptor with the script "/sbin/init.d/SnmpNaa" using the following command as root:

   nohup sh /sbin/init.d/SnmpNaa start

2. Start the Netegrity subagent with the script "/sbin/init.d/SnmpNetegrity" using the following command (run as root):

   nohup sh /sbin/init.d/SnmpNetegrity start

3. To stop the Netegrity subagent on HP-UX, run the following command as root:

   sh /sbin/init.d/SnmpNetegrity stop

Appendix A: Installation Worksheets

Use the worksheets in this section to help you record information that is required to install or upgrade SOA Security Manager.

Policy Server Worksheets

Use the following worksheets to record the necessary information to install the Policy Server and configure Policy Server components.

Required Information Worksheet

Use this worksheet to gather the required information before running the Policy Server installer.

Information Needed	Your Value
JRE location	
Policy Server installation location	
Encryption key value	

OneView Monitor Information Worksheet

If you plan on configuring the OneView Monitor, use this worksheet to gather information.

Information Needed	Your Value
JDK path	
ServletExec installation directory	
ServletExec port number	
Sun Java System administrator directory	
Multiple ServletExec instances	

ADAM Server Information Worksheet

Use this worksheet to gather the required information to configure an ADAM policy store.

Information Needed	Your Value
System IP address	
Server instance port number	
Root DN of the application partition	
ADAM administrator domain name	
ADAM administrator password	
Alternate LDAP administrator	
SOA Security Manager super user password	

Sun Java System Directory Server Information Worksheet

Use this worksheet to gather the required information to configure a Sun Java System Directory Server policy store.

Information Needed	Your Value
System IP address	
Directory instance port number	
Root DN	
Administrator account	
Administrator password	
Alternate LDAP administrator	
SOA Security Manager super user password	

SM Key Database Information Worksheet

If you plan on configuring the smkeydatabase, use this worksheet to gather the required information.

Information Needed	Your Value
smkeydatabase password	

Policy and Data Store Worksheets

You can use the following worksheets to record the necessary information to configure:

- An LDAP database as a policy store
- A relational database as a policy store
- An individual relational database as an audit logging database, key store, token store or session store

CA Directory Information Worksheet

You can use this worksheet to gather the required information for configuring a CA Directory database as a policy store.

Information Needed	Your Value
Host information	
CADSA port number	
Base DN	
Administrative DN	
Administrative password	

More information:

Gather Directory Server Information (see page 91)

Sun Java System Directory Server Information Worksheet

You can use this worksheet to gather the required information for configuring a Sun Java System Directory Server as a policy store or upgrading an existing policy store.

Information Needed	Your Value
Host information	
Directory server port information	
Administrative DN	
Administrative password	
Policy store root DN	
(Optional) SSL client certificate	

More information:

Gather Directory Server Information (see page 102)

Active Directory Information Worksheet

You can use this worksheet to gather the required information for configuring an Active Directory directory server as a policy store or upgrading an existing policy store.

Information Needed	Your Value
Host information	
Directory server port information	
Administrative DN	
Administrative password	
Policy store root DN	
(Optional) SSL client certificate	

More information:

Gather Directory Server Information (see page 102)

Microsoft ADAM Information Worksheet

You can use this worksheet to gather the required information for configuring a Microsoft ADAM directory server as a policy store or upgrading an existing policy store.

Information Needed	Your Value
Host information	
Directory server port information	
Administrative DN	
Administrative password	
Policy store root DN	
(Optional) SSL client certificate	

More information:

Gather Directory Server Information (see page 122)

SQL Server Information Worksheet

Use this worksheet to gather the required information for configuring a SQL Server database to function as a policy store or any other type of SOA Security Manager data store.

Information prefixed with (W) indicates that the information is only required if the Policy Server is installed on a Windows system; (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. Different information is required when configuring the SQL Server data source.

Information Need	Your Value
Database instance name	
Database administrative account	
Database administrative password	
(W) Data source name	
(W) SQL Server name	
(U) Policy Server root	
(U) IP address	

Oracle Information Worksheet

Use this worksheet to gather the required information for configuring an Oracle database to function as a policy store or any other type of SOA Security Manager data store.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring the Oracle data source for UNIX.

Information Needed	Your Value
(U) Policy Server installation path	
Data source	
Database administrative account	
Database administrative password	
Oracle machine name	
Oracle instance service name	
Oracle port number	

Oracle RAC Information Worksheet

Use this worksheet to gather the required information for configuring an Oracle RAC database to function as a policy store or any other type of SOA Security Manager data store.

Information prefixed with (U) indicates that the information is only required if the Policy Server is installed on a UNIX system. This information is required when configuring the Oracle data source for UNIX.

Information Needed	Your Value
(U) Policy Server installation path	
Data source name	
Database administrative account	
Database administrative password	
Oracle RAC system service name	
Oracle RAC node service names	
Oracle RAC node IP addresses	

Information Needed	Your Value
Oracle RAC node port number	

Administrative UI Installation Worksheets

You can use the following worksheets to record the necessary information to:

- Install the Administrative UI.
- Configure the connection between the Administrative UI and a Policy Server.

More information:

Gather Application Server Information (see page 278)

JBoss Worksheet

You can use this worksheet to gather the required JBoss information for the Administrative UI installation:

Required Information	Your Value
JBoss installation folder	
JBoss URL	
JDK	

More information:

JBoss Information (see page 279)

WebLogic Worksheet

You can use this worksheet to gather the required WebLogic information for the Administrative UI installation:

Required Information	Your Value
WebLogic binary folder	
WebLogic domain folder	
WebLogic server name	

Required Information	Your Value
Application server URL and port	

WebSphere Worksheet

You can use this worksheet to gather the required WebSphere information for the Administrative UI installation:

Required Information	Your Value
WebSphere installation folder	
WebSphere URL	
Server name	
Profile name	
Node name	
Cell name	
JDK	

More information:

WebSphere Information (see page 280)

Object Store Worksheet

You can use this worksheet to gather the required object store information for the Administrative UI installation:

Required Information	Your Value
Host name	
Object store name	
Object store port number	
Object store administrator account	
Object store administrator password	

More information:

Gather Object Store Information (see page 280)

LDAP Administrative User Store Worksheet

You can use this worksheet to gather the required LDAP administrative user store information for the Administrative UI installation:

Required Information	Your Value
Port number	
Root DN	
Directory manager DN	
Directory manager password	
Admin Super User DN	
Directory.xml file location	

More information:

LDAP Administrator User Store Information (see page 281)

ODBC Administrative User Store Worksheet

You can use this worksheet to gather the required ODBC administrative user store information for the Administrative UI installation:

Required Value	Your Value
Host name	
Database port	
Database name	
Database administrator account	
Database administrator password	
Admin Super User	
Directory.xml file location	

More information:

ODBC Administrator User Store Information (see page 282)

Administrative UI Registration Worksheet

You can use this worksheet to gather the required registration information for the Administrative UI installation:

Required Information	Your Value
Client name	
Passphrase	
Policy Server host name	
Policy Server port number	

More information:

Gather Registration Information (see page 292)

Appendix B: Troubleshooting

This section contains the following topics:

Policy Server Troubleshooting

The following sections detail common problems you may experience with the Policy Server during installation and the proposed solutions.

NETE_PS_ALT_CONF_FILE Environment Variable on Solaris

After installing the Policy Server on Solaris, the nete_ps_env.ksh script may have the following entry:

```
export NETE_PS_ALT_CONF_FILE=/export/siteminder/config/.siteminder.conf
```

The NETE_PS_ALT_CONF_FILE environment variable is used by the stop-all and start-all scripts, which stop and start the Policy Server's service. The .siteminder.conf file is a temporary, run-time file created by these scripts and has no affect your SOA Security Manager configuration.

Do not modify the NETE_PS_ALT_CONF_FILE environment variable.

Policy Server Fails to Start After Installation

Valid on Windows and UNIX Systems

Symptom:

I have installed the Policy Server, but it is not starting.

Solution:

You may have the wrong JRE version installed. Make sure you have the correct JRE version.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site.

Winsock error 10054 message

Valid on Windows

Symptom:

When I try to log into the Policy Server, I receive the "Unable to proceed, winsock error 10054" message.

Solution:

One of the following could be the cause of the problem:

- The policy store does not contain the proper SOA Security Manager schema. Make sure you imported the correct SOA Security Manager schema.
- The Policy Server is not running. To start this server, use the Status tab on the Policy Server Management Console.

  Important! If you are accessing this graphical user interface on Windows Server 2008, open the shortcut with Administrator permissions, even if you are logged into the system as an Administrator. For more information, see the release notes for your SOA Security Manager component.
- The Policy Server is not connected to the policy store properly. Using the Data tab on the Policy Server Management Console, click Test Connection to make sure the policy store connects successfully. If it does not, reenter the data source information values on the Data tab by pointing the Policy Server at the policy store.

More Information:

Relational Databases as a Policy or Key Store (see page 131)

Policy Store Troubleshooting

The following sections detail common problems you may experience with the policy store during installation and the proposed solutions.

Policy Stores with Large Numbers of Objects

Valid on Windows and UNIX Systems

Symptom:

My Policy store has returned the exception java.lang.IndexOutOfBounds to the FSS Administrative UI.

Solution:

Policy Stores with large numbers of objects may return the exception java.lang.IndexOutOfBounds to the Administrative UI.

Define the registry key \HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\ObjectStore\MaxObjects to a value lower than 100 (such as 50).

SSL initialization failed: error -8174 (security library: bad database.)

Valid on Windows and UNIX Systems

Symptom:

When I run smldapsetup ldmod -fpstore -ssl1 -c/app/siteminder/ssl/cert7.db for policy stores that are using an SSL-encrypted connection to Sun Java System Directory Server, I receive the following error message:

"SSL initialization failed: error -8174 (security library: bad database.)"

Solution:

1. Make sure the key3.db file exists in the same directory as cert7.db for the Netscape Web browser.
2. Rerun this smldapsetup command, and, for the -c option, specify the path of the directory where the SSL client certificate database file, cert7.db, exists.

   Example: if cert7.db exists in /app/siteminder/ssl, specify -c/app/siteminder/ssl/cert7.db

ODBC Policy Store Import Fails with UserDirectory Error

Symptom:

I receive an error message stating that the policy store failed operation "save" for object type "UserDirectory" when importing policy store data into an ODBC policy store.

Solution:

It is possible that the server name in the ODBC store's userDirectory object is longer than 512 characters, which by default, exceeds the limit allowed by the MS SQL Server and Oracle policy store schema scripts that are shipped with SOA Security Manager.

Do one of the following:

If you are trying to import policy data into a supported version of a MS SQL Server policy store:

1. Open sm_mssql_ps.sql.

   Note: This schema script is located in *policy_server_home*\db\SQL.

2. Search for the following text:

   CREATE TABLE smuserdirectory5

3. Modify "server smstringreq512N," to one of the following depending on your needs:

   - server smstringreq1024,N
   - server smstringreq4000,N

4. Re-import the policy store schema into the policy store.
5. Import the policy store data.

If you are trying to import policy data into a supported version of an Oracle policy store:

1. Open sm_oracle_ps.sql.

   Note: This schema script is located in *policy_server_home*\db\SQL.

2. Search for the following text:

   CREATE TABLE smuserdirectory5

3. Modify "server VARCHAR2(512) NOT NULL," to one of the following depending on your needs:

   - server VARCHAR2(1024) NOT NULL,
   - server VARCHAR2(4000) NOT NULL,

4. Re-import the policy store schema into the policy store.
5. Import the policy store data.

OneView Monitor Troubleshooting

The following sections detail common problems you may experience with the OneView Monitor during installation and the proposed solutions.

Fix Modified UNIX/Sun Java System Web Server Configuration Files

As mentioned in the procedure for installing the ServletExec/AS on a UNIX/Sun Java System, we advise not allowing ServletExec to modify the Sun Java System Web server's configuration files (obj.conf and magnus.conf). However, if ServletExec did modify these files during installation, the Web server instance fails after you configure the FSS Administrative UI and OneView Monitor GUI using the Policy Server installer/wizard. The ServletExec installer puts entries in these files that conflict with those from the Policy Server.

To keep the Web server instance from failing, remove the conflicting entries from the Sun Java System Web Server instance's obj.conf and magnus.conf files.

1. Open /<*sunjavasystem_home*>/servers/https-<*web_server_instance_name*>.domain.com/config/magnus.conf, and remove the first line:

   Init fn="ServletExecInit" <*ServExec_instance_name*>.instances="<*IP_address*>:<*portnumber*> "

   ServExec_instance_name

   Specifies the name of your SerlvetExec instance.

   IP_address

   Specifies the IP Address of the machine where ServletExec is installed.

   portnumber

   Specifies the port number for the SerlvetExec instance.

   Note: The Policy Server Configuration Wizard added the correct entry at the end of the file.

2. Open /<*sunjavasystem_home*>/servers/https-<*web_server_instance_name*>.domain.com/config/obj.conf, and remove lines four and five from the top of the file:

   NameTrans fn="assign-name" from="/servlet/*" name="<*ServExec_instance_name*>"
   NameTrans fn="assign-name" from="*.jsp*" name="<*ServExec_instance_name*>"

Important! Do not remove the name="se-<ServExec_instance_name>" entries in lines two and three since these were added by the Policy Server Configuration Wizard.

3. In the same obj.conf file, remove the second to the last <Object *name="<ServExec_instance_name>*"> section from the end of the file:

```
<Object name="<ServExec_instance_name>">
Service fn="ServletExecService" group="<ServExec_instance_name>"
</Object>
```

Important! Do not remove the <Object name="se-<ServExec_instance_name>"> entry since this one was added by the Policy Server Configuration Wizard.

4. After saving these files, you should be able to start the Web server instance from the Sun Java System Web Server Administration Server page.

Windows/IIS Virtual Path to /siteminderdermonitor Does Not Exist

Valid on Windows

Symptom:

The virtual path to the /siteminderdermonitor does not exist under Default Web Site in the IIS Microsoft Management Console.

Solution:

Create the virtual path.

To create the virtual path

1. From the Start menu, go to: Programs, Administrative Tools, Internet Service Manager.
2. Select Default Web Site.
3. From the Action menu, select New, Virtual Directory.

   The Virtual Directory Wizard opens.
4. Specify the name (alias) of the virtual directory. For example: siteminderdermonitor

   Note: You can specify any name for the alias as siteminderdermonitor is an example
5. Click Next.
6. Specify the path to <*siteminder_installation*>\monitor\.
7. Click Next.

8. Select the Allow Execute Access permission.
9. Click Finish.

Administrative UI Troubleshooting

The following sections detail common problems you may experience with registering the Administrative UI and the proposed solutions.

HTTP Status: 404 Error Appears

Problem:

I receive an HTTP Status: 404 error message when I try to log in to the Administrative UI.

Consider the following solutions:

Solution 1:

If you are trying to register the Administrative UI with the Policy Server, some or all of the administrator user store credentials you entered when installing the Administrative UI are incorrect.

1. Verify the following information:
   - **Administrator credentials**—The directory manager credentials (LDAP) or the database administrator credentials (ODBC) used to connect to the administrator user store may be incorrect.
   - **Administrative UI Super User**—The default Super User's user DN (LDAP) or account name (ODBC) may be incorrect.
2. Re-install the Administrative UI with the correct administrator user store credentials.

   Note: If you have not exceeded the registration timeout value, you do not have to run XPSRegClient again to create a client name and passphrase. The default timeout value is four (4) hours.

Solution 2:

The object store Administrative password may have changed. If the password has changed, update the application server's datasource definitions with the new datasource password.

To update the datasource password for WebLogic or WebSphere, use the vendor-specific administrative tool.

Note: See you vendor-specific documentation for more information on updating datasource passwords.

To update the datasource password for JBoss

1. Obtain the new password.
2. Navigate to one of the following in a console window:
   - (Windows) *admin _ui_home*\CA\IAM Suite\siteminderWAM\tools \PasswordTool

     admin_ui_home

     Specifies the Administrative UI installation path.
   - (UNIX) /opt/CA/IAM_SUITE/siteminderWAM/tools/PasswordTool
3. Run the following command:
   - (Windows) pwdtools -JSAFE -p *password*
   - (UNIX) ./pwdtools.sh -JSAFE -p *password*

   password

   Specifies the Administrative password for the object store.

   The password utility encrypts the password.
4. Copy the encrypted password.
5. Navigate to *jboss_home*\server*server_name*\conf and open the login-config.xml file.

   jboss_home

   Specifies the JBoss installation path.

   server_name

   Specifies the name of the JBoss server.
6. Locate the following policy application sections in the login-config.xml file:
   - <application-policy name="objectstore-login">
   - <application-policy name="reportsnapshot-login">
   - <application-policy name="imtaskpersistencedb-login">
   - <application-policy name="imauditdb-login">

   Each of these sections includes a <module-option name="password"> parameter.
7. Replace the existing value of each instance of the module-option name parameter with the new encrypted value.
8. Save the file.
9. Restart JBoss.

API Error Appears

Symptom:

The Administrative UI registration fails with an Agent API failure message.

Solution:

The Policy Server is not started. Start the Policy Server using the Policy Server Management Console.

Registration Not on File Error Appears

Symptom:

The Administrative UI registration fails with a registration record not on file error message.

Solution:

Do the following:

1. Verify that the client name you entered is identical to the client name you created using XPSRegClient. The value you created must match the value that you enter using the Administrative UI. If you do not have a client name, proceed to step 2.
2. Run XPSRegClient to create a client name and passphrase. This is required information to complete the Administrative UI registration process.

Invalid Registration File Error Appears

Symptom:

The Administrative UI registration fails with an invalid registration file error message.

Solution:

Do the following:

1. Verify that the passphrase you entered is identical to the passphrase you created using XPSRegClient. The value you created must match the value that you enter using the Administrative UI. If you do not have a passphrase, proceed to step 2.
2. Run XPSRegClient to create a client name and passphrase. This is required information to complete the Administrative UI registration process.

Registration Fails without Timeout

Symptom:

The Administrative UI registration fails without timing out.

Solution:

Do the following:

- Ping the machine hosting the Policy Server to be sure it is available.
- Locate the following registration file in *policy_server_home*\bin:

  name.XPSReg

  policy_server_home

  Specifies the Policy Server installation path

  name

  Identifies the client name you specified when using the Administrative UI registration tool (XPSRegClient) to create a client name and passphrase.

  If the registration file does not exist, run XPSRegClient to create a client name and passphrase.
- Open the Policy Server log file (smps.log) and review it for errors that may have occurred around the time of the registration.

No Tabs Appear in the Administrative UI

Symptom:

No tabs appear when I log into the Administrative UI for the first time to register it with a Policy Server.

Solution:

Although you have logged in using the credentials of a user in the administrator user store, this is not the individual you identified as the Super User when installing the Administrative UI. Log in using the credentials of the user you identified as the Super User when installing the Administrative UI.

Additional Tabs do not appear after Registration

Symptom:

I have successfully registered the Administrative UI with a Policy Server, but the Infrastructure and Policies tabs do not appear.

Solution:

It is possible that you did not use the -su argument when using the Administrative UI registration tool (XPSRegClient) to create the client name and passphrase.

1. Re–run the Administrative UI registration tool with the -su argument.

   Note: Be sure that you specify a unique client name. For example, if you had previously specified smui1 as a client name, specify smui2.

2. Configure the connection between the Administrative UI and Policy Server.

Cannot Find the Administrative UI Registration Log

Symptom:

I am trying to troubleshoot the Administrative UI registration and cannot find the log file.

Solution:

XPSRegClient creates and saves the log file in *policy_server_home*\log. The file name is XPSRegClient.*date*

policy server home

Specifies the Policy Server installation path.

date

Specifies the date on which XPSRegClient created the file.

Example: XPSRegClient.2007-12-1.154002

Note: The last six digits are a unique identifier you can use if more than one file is created on the same day.

Search Fails with Timeout Error

Symptom:

I cannot complete a search for policy objects. The Administrative UI displays a connection timeout error instead of returning the search results.

Solution:

When searching on a large number of policy objects via the Administrative UI, the connection between the Administrative UI and the Policy Server may time out and/or the Policy Server tunnel buffer may become corrupt. This results in a connection timeout error. Adjusting the Administrative UI Policy Server connection timeout and creating a registry key for the Policy Server tunnel buffer size solves this problem.

To adjust the Policy Server connection timeout

1. Log in to the Administrative UI.
2. Click Administration, Admin UI, Policy Server Connections, Modify Policy Server Connection, Search to open the Policy Server connection object.
3. Select the appropriate Policy Server and click Submit.
4. Set the Timeout field in the Advanced section to a large value, such as 2,000 seconds.

   The Policy Server connection timeout is now increased.

To create a registry key for the tunnel buffer size

1. Create the following Policy Server registry key:

   HKLM\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\PolicyServer\

   Max AdmComm Buffer Size
2. Set this registry key to a large value, such as 5,910,496 bytes (0x5a2fe0).
3. Save the changes and exit the registry.

Note: If the problem persists after the connection timeout and buffer size changes, restart the Administrative UI

Default Log File does not Provide Enough Information

Symptom:

I am trying to troubleshoot the Administrative UI installation and the default log file does not provide enough information.

Solution:

When you start the Administrative UI Task Engine, by default, output is sent to the default log file. The location of the default log file is dependent on the type of application server on which the Administrative UI is running.

- **Jboss**—*jboss_home*/server/*server_name*/log/server.log

  jboss_home

  Specifies the Jboss installation path.

  server_name

  Specifies the Jboss server name.

- **WebSphere**—*websphere_home*/AppServer/profiles/*profile_name*/logs/*server_name*/SystemOut.log

  websphere_home

  Specifies the WebSphere installation path.

  profile_name

  Specifies the WebSphere profile name.

  server_name

  Specifies the WebSphere server name.

- **WebLogic**—WebLogic logs are available on the WebLogic console.

You can use the SiteMinderLog4j.properties file, which is located in the *deploy*/IdentityMinder.ear/user_console.war/META-INF to configure SOA Security Manager-specific logging settings.

deploy

Specifies the location of where your application server deploys applications.

Note: More information on how to configure the SiteMinderLog4j.properties file is commented in the file.

FSS Administrative UI Troubleshooting

The following sections detail common problems you may experience with using the FSS Administrative UI and the proposed solutions.

FSS Administrative UI Fails to Start in IE

Symptom:

When I attempt to start the FSS Administrative UI in Internet Explorer it fails.

Solution:

Make sure that the Java Plug-in is set as the default Java Runtime in the browser by doing the following:

1. From the Control Panel, select Java Plug-In.
2. On the Browser tab, make sure that Microsoft Internet Explorer is checked and click Apply.
   - If the plug-in is not set, the FSS Administrative UI stalls indefinitely at the Downloading Administration dialog box.
   - If you are still have difficulty getting the FSS Administrative UI to display, run the FSS Administrative UI Browser Compatibility Test at the following URL: http://www.netegrity.com/uitest
   - If the panel is a solid box, click Details for more troubleshooting information.

FSS Administrative UI does not appear on Windows

Valid on Windows

Symptom:

I have installed the FSS Administrative UI, but it does not appear.

Solution:

If you installed an Internet Information Server (IIS) Web Server on a port other than 80, the FSS Administrative UI may not appear. Editing the FSS Administrative UI shortcut URL should fix the problem.

To edit the shortcut URL.

1. Right-click Start and select Open All Users.
2. Double-click Programs.
3. Double-click SOA Security Manager.
4. Highlight the FSS Administrative UI icon, right click, and select Properties.
5. Select the Web Document tab.

6. In the URL field on the Web Document tab, change the port number to the one you configured for the IIS Web Server.

   Example: if your IIS Web Server is located on port 81, change:

   http://<*fully qualified domain name*>:80/siteminder/index.htm

   to http://<*fully qualified domain name*>:81/siteminder/index.htm

7. Click Apply and OK.

If you are still have difficulty getting the FSS Administrative UI to display, run the FSS Administrative UI Browser Compatibility Test at the following URL: http://www.netegrity.com/uitest

If the panel is a solid box, click Details for more troubleshooting information.

FSS Administrative UI Fails to Start on a Sun Java Web Server

Symptom:

When I attempt to start the FSS Administrative UI that is configured for a Sun Java System Web server, it fails to start.

Solution:

Disable the Java Enabled Globally option by doing the following:

1. Open the Sun Java System Enterprise Administration Server home page by entering the following URL in a browser:
   http://<yourserver.com>:<webserverport>

   yourserver.com

   Specifies the domain name of the Enterprise Administration Server.

   webserverport

   Specifies the port number.

2. In the Select a Server drop-down menu, select a server, and then click Manage.
3. Select the Java tab.
4. Deselect Enable Java for class defaultclass and Enable Java Globally and click OK.
5. Turn on the Web server.

Java Error Messages When Uninstalling

Symptom:

When I attempt to uninstall the Policy Server, Web Agent, SDK, or the documentation, the uninstallation program stops and issues one of the following error messages:

- "Could not find a valid Java virtual machine to load. You need to reinstall a supported Java virtual machine."
- "No Java virtual machine could be found from your PATH environment variable. You must install a VM prior to running this program."

Solution:

Make sure the JRE is in the PATH variable.

Set the JRE in the PATH Variable on Windows

To Set the JRE in the PATH variable

1. Go to the Control Panel.
2. Double-click System.
3. In the Environment Variables dialog, add the location of the JRE to the Path system variable.

Set the JRE in the PATH Variable on Solaris

To set the JRE in the PATH variable

Run the following commands:

1. PATH=$PATH:<JRE>/bin

   JRE

   Specifies the location of your JRE.

2. export PATH

Adobe Acrobat Reader Won't Install

Valid on Windows

Symptom:

When I try to install Adobe Acrobat, the installation program hangs.

Solution:

If the Acrobat Reader installation program hangs while the Policy Server service is running, stop it using the Policy Server Management Console's Status tab. After stopping the service, the installation program should start.

Important! If you are accessing this graphical user interface on Windows Server 2008, open the shortcut with Administrator permissions, even if you are logged into the system as an Administrator. For more information, see the release notes for your SOA Security Manager component.

Problem With Using Active Directory as a User Store

Symptom:

When I use Active Directory as a user store, the Policy Server issues error messages that it cannot connect to this store.

Solution:

When creating an Active Directory-based user store, make sure you specify a fully qualified host name (for example, host.domain.com) in the Administrative UI and do not use the machine's IP Address. Moreover, make sure you can ping host.domain.com and domain.com from the machine where the Policy Server is installed since Active Directory sends referrals to the Policy Server that are identified by the fully qualified host name. If the fully qualified host names are invalid and unreachable, the Policy Server issues error messages.

AE failed to load library 'smjavaapi'. System error

Valid on Windows and UNIX Systems

Symptom:

During Authorization, I receive the "AE failed to load library 'smjavaapi'. System error: The specified module could not be found." error message.

Solution:

Set the PATH variable to <*SiteMinder_installation*>\config\JVMOptions.txt for Windows or the LD_LIBRARY_PATH to <*SiteMinder_installation*>/config/JVMOptions.txt for UNIX systems.

Appendix C: XA on MS SQL 2005

This section contains the following topics:

How to Enable XA on MS SQL 2005

When using WebSphere with Microsoft SQL 2005, enable XA transactions. Because WebSphere does not exclude data sources from Global transactions, we must perform transaction management that spans across databases. Also, when a workflow process initiates transactions, the call comes from an Enterprise Java Bean, therefore all data sources that interact with this call must be participants of the transaction.

To enable XA, perform the following steps on the SQL Server system:

1. Confirm that the Distributed Transaction Coordinator service is running. (see page 351)
2. Install the stored procedures for the Java Transaction API (JTA) (see page 352).
3. Create a new registry-named value (see page 352).
4. Enable XA transactions (see page 352).

Confirm the Distributed Transaction Coordinator is Started

Perform the following procedure on each system where Microsoft SQL Server is installed.

To confirm the Distributed Transaction Coordinator service is running

1. Go to Start, Control Panel, Administrative Tools, Services.
2. Verify that the Distributed Transaction Coordinator (DTC) is started.
3. If the DTC is not started, start the DTC service.

Install Stored Procedures for JTA

1. MicrosoftSQL Server service.
2. Run the instjdbc.sql script.

   The instjdbc.sql script can be run by the Microsoft SQL Management Studio or the ISQL utility.

   Note: The first time you run this script, you can expect to see errors about removing stored procedures.

Create a New Registry-Named Value

To create a new registry-named value

1. Using the Registry Editor, navigate to the following registry key:
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSDTC\XADLL
2. Go to Edit, New, String Value.
3. Create a new registry-named value with the following parameters:

   Name: sqljdbc.dll

   Type: String (REG_SZ)

   Value: *sqlserver_home*\Binn\sqljdbc.dll

   sqlserver_home

   Specifies the Microsoft SQL Server installation location.

Enable XA Transactions

To enable XA transactions

1. Go to Start, Control Panel, Administrative Tools, Component Services.
2. Under Computers, navigate to the system where you want to enable XA transaction support.
3. Right click the system name, click Properties.
4. Under the MSDTC tab, click Security Configuration.
5. Select the check box for Enable XA Transactions.
6. Click Ok.
7. (Optional) If you have a database cluster, repeat Step 2 for every SQL server in the cluster.

Appendix D: Configuring the Policy Server for an International Environment

This section contains the following topics:

Policy Servers in an International Environment

The Policy Server supports SOA Security Manager data stores residing in an Oracle or SQL Server database, and LDAP servers for an international environment.

Planning Considerations Before Installing the Policy Server

Consider the following before installing the Policy Server:

- Use supported operating system and third-party software
- Create supported databases
  - Before creating databases for storing policy or session data, make sure they are formatted with UTF-8 encoding.
  - User store databases are not limited to UTF-8 encoding. User databases may be created in the local character set encoding.

    Example: to store Japanese characters in Oracle 9i, the database can be created in JA16EUC encoding or JA16SJIS encoding.

    Note: The Active Directory namespace does not support multi-byte characters. Regardless of the code page you are using, SOA Security Manager treats characters as they are defined in Unicode. Although your code page may reference a special character as single-byte, SOA Security Manager treats it as a multi-byte character if Unicode defines it as such.
- All Administrative UI fields support multi-byte characters.
- Some of the FSS Administrative UI fields support multi-byte characters.

- Some Policy Server components support multi-byte and ASCII characters in an internationalized environment.
- SOA Security Manager supports multi-byte character (MBCS) URLs.

Note: For a list of supported CA and third-party components, refer to the SOA Security Manager r12.1 Platform Support Matrix on the Technical Support site.

User Interface Fields Supporting Multi-byte Characters

The Administrative UI and FSS Administrative UI contain fields that support multi-byte characters. The following sections detail the specific fields.

Administrative UI

All of the fields in the Administrative UI support multi-byte characters.

Federation Security Services Administrative UI

The following FSS Administrative UI dialog box fields support multi-byte characters:

Field Name	Dialog Where Field is Located	Access Dialog By Selecting:
All Description fields	All dialogs	Any dialog that has a Description field
All Name fields except dialogs listed in next column	All dialogs except Name fields in: ■ SOA Security Manager Agent Dialog ■ SOA Security Manager Agent Group Dialog ■ SOA Security Manager Agent Configuration Object Dialog ■ SOA Security Manager Host Configuration Object Dialog ■ SOA Security Manager Agent Type Dialog	Access Dialog by selecting: ■ Edit, Create Agent ■ Edit, System Configuration, Create Agent Group ■ Edit, System Configuration, Create Agent Conf Object ■ Edit, System Configuration, Create Host Conf Object ■ Edit, System Configuration, Create Type
DN field under the User Session Caches group	SOA Security Manager Cache Management Dialog	Tools, Manage Cache
Value and Enter Single Value fields	SOA Security Manager Host Configuration Object's Edit Parameter Dialog	Edit, System Configuration, Create Host Conf Object, click Add

Root, Start, and End fields for the LDAP and AD namespaces	SOA Security Manager User Directory Dialog	Edit, System Configuration, Create User Directory
Username field	SOA Security Manager User Directory Dialog's Credentials and Collection tab	Edit, System Configuration, Create User Directory
Filter (binoculars) fields	SOA Security Manager Policy Domain Dialog	Edit, System Configuration, Create Domain
Manual Entry field in the Condition group Value field in the Infix Notation group	SOA Security Manager Password Policy Dialog's SOA Security Manager User Lookup screen	Edit, System Configuration, Create Password Policy
Function Parameter field of the SOA Security Manager Active Rule editor dialog	SOA Security Manager Rule and SOA Security Manager Global Rule dialogs	Click a realm and select Create Rule Under Realm
■ Variable Value field in the Static option. ■ DN Spec field in the DN Attribute option. ■ Parameters field in Active Response option.	SOA Security Manager Response and Global Response dialogs	Click a response and select Create Response
■ On the Users tab, the Manual Entry field in Condition group and the Value field in Infix Notation group. ■ On the Rules tab, the Filter fields (binoculars) of the Set Response and Set Global Response dialogs. ■ On the Advanced tab, the Function Parameter field in Active Policy group.	SOA Security Manager Policies and Global Policies dialogs	Click Policies and select Policy
■ Value field in Attribute-Value option. ■ Search Expression field in the LDAP Query option.	SOA Security Manager User Lookup screen	In multiple dialogs

Policy Server Components Supporting Multi-byte Characters

The following Policy Server components support multi-byte and ASCII characters in an internationalized environment:

- Administrative UI
- Policy Server Management Console
- Authentication Schemes
  - HTML Forms
  - X.509 Client Certificate
  - X.509 Client Certificate and HTML Forms
  - X.509 Client Certificate or HTML Forms
  - RADIUS CHAP/PAP
  - SecurID Authentication
  - Anonymous Authentication
  - Custom Authentication
  - Impersonation Authentication
- Password Services

  Note: Password Services are limited to ASCII characters, but can support a multi-byte character URL as a redirection URL.
- Responses
- Post Preservation
- SiteMinder Test Tool
- Audit logging to text files
- Audit logging to ODBC databases
- smobjexport and smobjimport
- XPSExport and XPSImport
- Java Agent API

Support for Multi-Byte Character URLs

SOA Security Manager supports URLs that contain multi-byte characters (MBCS). MBCS URL support includes support for:

- **Internationalized domain names** - An *internationalized domain name* (IDN) is an Internet domain name that can contain non-ASCII characters, including letters with diacritics and characters from non-Latin scripts, such as Arabic and Chinese.
- **Internationalized resource identifiers** - An *internationalized resource identifier* (IRI) is the international equivalent of a uniform resource identifier (URI). An IRI can contain ASCII characters and characters from a MBCS set; a URI is limited to a subset of ASCII characters.

MBCS URL support lets:

- SOA Security Manager protect resources that are accessed through MBCS URLs.
- You configure specific authentication schemes using an IDN and an IRI.

How to Enable MBCS URL Support

Support for MBCS URLs in a SOA Security Manager environment requires that:

- The Web browsers used to access the protected resources meet specific requirements.
- The Web server implementation in your environment meets specific requirements.
- Specific default bad characters are removed from the Web Agent Configuration Object.

To enable support for MBCS URLs:

1. Ensure that the Web browsers meet the requirements for MBCS URLs.
2. Ensure that the Web servers meet the requirements for MBCS URLs.
3. Configure the Web Agent Configuration Object.

Web Browser Requirements for MBCS URLs

Web browsers must be able to send requests to Web servers that serve resources in UTF-8 format and whose domain names contain non-ASCII characters.

The Web browsers used to access the protected resources must be able to:

- Support Internationalized Domain Names (IDNs).
- Support Internationalized Resource Identifiers (IRIs).
- Send requests in UTF-8 format.

Web Server Requirements for MBCS URLs

A Web server can support MBCS URLs if it meets at least one of the following requirements:

- The Web server can convert UTF-8 requests to the local character set encoding.

  or

- The Web server can store files in UTF-8 format. This lets the Web server serve the file when it receives the IRI request from the Web browser in UTF-8 format.

Enable Multi-byte Character Support

MBCS support requires that you remove specific high-bit ASCII character values from the Web Agent Configuration Object.

Note: Removing the high-bit ASCII characters prevents the Web Agent from blocking the specific characters.

To enable MBCS support

1. Open the Administrative UI
2. Click Infrastructure, Agents.
3. Click Agent Configuration, Modify Agent Configuration.

   The Modify Agent Configuration pane appears.
4. Enter search criteria and click Search.

   Agent configuration objects matching the search criteria appear.
5. Select the Agent configuration object you want and click Select.

   Agent Configuration parameters are listed in the Parameters group box.
6. Click the Edit icon for BadURLChars.

   The Edit Parameter pane appears.
7. Remove the following from the Values field:
   - %00-%1f
   - %7f-%ff

8. Click OK.

   The edited values appear in the BadURLChars field.

9. Click Submit.

   The Web Agent Configuration Object is configured to support MBCS URLs.

Protect a Resource with MBCS URLs

Support for MBCS URLs lets SOA Security Manager protect resources that are accessed through URLs that contain non-ASCII characters.

When creating a realm and the associated rule or rules to protect the resource, you can enter a MBCS URL in the Resource field. Users can access the protected resource using a browser that supports IDNs and IRIs.

Note: More information on creating realms and rules exists in the *Policy Server Configuration Guide*.

Authentication Schemes Supporting MBCS URLs

You can configure the following authentication schemes with an IDN in the Server Name field and an IRI in the Target field:

- Basic over SSL
- HTML Form Template
- HTML Form Template over SSL
- X509 Client Cert
- X509 Client Cert and Forms

Note: Netscape and Firefox do not accept redirections to URLs that contain an IDN. Entering an IDN for a forms-related authentication scheme results in a failure unless Punny code is used. More information on configuring authentication schemes exists in the *Policy Server Configuration Guide*.

Configure SOA Security Manager Data Stores Supporting International Characters

You can configure SOA Security Manager data stores in SQL Server or Oracle databases. When configuring these data stores, be aware that the Policy Server only supports UTF-8 encoding and, as a result, you must use databases that support this encoding type.

Note: This section applies to configuring SOA Security Manager data stores in relational databases. More information on configuring these stores in LDAP servers exists in LDAP Directory Servers as a Policy Store or Key Store.

Configure an International SOA Security Manager Data Store in SQL Server

To create policy, keys, session, or key stores, configure a SOA Security Manager data store in the SQL Server database.

Note: By default, SQL Server supports UTF-8 character encoding.

Configure an International SOA Security Manager Data Store in Oracle

To configure an international SOA Security Manager data store in Oracle

1. On the machine where Oracle is installed, create a custom Oracle database that supports UTF-8 character encoding.

   Note: For more information and instructions, see Oracle's documentation.

   To verify if an existing Oracle database supports UTF-8 character encoding, run the following query:

   Select * from nls_database_parameters where parameter = 'NLS_CHARACTERSET'

2. Create policy, keys, session, or key stores for the Policy Server, by configuring a SOA Security Manager data store in the Oracle database.

Solaris/LINUX Red Hat Policy Server Logging UTF-8 Characters to an Oracle Database

A Solaris/LINUX Red Hat Policy Server can log UTF-8 characters to an Oracle audit log database. To enable this configuration, you need to set the following environment variables:

For a simplified Chinese operating system

- LANG=zh_CN.utf8

For a Japanese operating system

- LANG=jp_JP.UTF-8

You set the LANG variable system-wide or just for the Policy server process.

Note: To avoid impacting any other applications, make sure that you set this variable for the Policy Server process only.

Database Driver Variable

- IANAAppCodePage=utf-8

You set this variable in the appropriate data source definition section of the system_odbc.ini file, installed in *<policy_server_installation>*/db.

Oracle Client Settings

Since the Policy Server uses the Oracle wire protocol driver, an Oracle client is not necessary. However, if you need an Oracle SQLPLUS client in your environment to read data from the audit log database, you may have to set one or both of the following environment variables to correctly display the multi-bytes characters:

For a simplified Chinese operating system

- LANG=zh_CN.utf8

For a Japanese operating system

- LANG=jp_JP.UTF-8

For the Oracle SQLPlus Client

- NLS_LANG (For example, NLS_LANG=Japanese_Japan.UTF8)

Note: For more information, see the operating system and database client configuration manual.

Configure a Japanese User Store in SQL Server

Using the smsampleusers_sqlserver.sql file installed with the Policy Server, you can configure a user store in a SQL Server database. This file is installed in the *<siteminder_installation>*\db\SQL directory.

Note: User stores are not limited to UTF-8 format. You can create a user store in the local character set encoding.

To configure a Japanese user store in SQL Server

1. Edit the smsampleusers_sqlserver.sql file, by doing the following:
   a. Replace every varchar instance with **nvarchar**.
   b. Place an **N** before any insert statement with international strings.

      Japanese example:

      insert into SmUser (UserID , Name, Password,
      LastName, FirstName, ...)

      values (12, **N**' やまもと ',
      'siteminder','guest','guest','guest@mycompany.com...)
2. Import the smsampleusers_sqlserver.sql file.

   Note: More information on importing the smsampleusers_sqlserver.sql file exists in Sample User Directories.
3. Open the Policy Server User Interface's SiteMinder ODBC Query Scheme dialog and modify the policy store's SQL query scheme by placing an **N** before every %s reference in any = %s statement.

   Example: the following sample query scheme statements:

   select Name, 'User' from SmUser where Name = '%s' Union select Name, 'Group' from SmGroup where Name = '%s'

   should become:

   select Name, 'User' from SmUser where Name = **N**'%s' Union select Name, 'Group' from SmGroup where Name = **N**'%s'
4. Stop and restart the Policy Server.

   The user store configuration is complete and now supports multi-byte characters.

Configure a Japanese User Store in Oracle

Using the smsampleusers_oracle.sql file installed with the Policy Server, you can configure a user store in an Oracle database. This file is installed in the *<siteminder_installation>*\db\SQL directory.

Note: User stores are not limited to UTF-8 format. You can create a user store in the local character set encoding.

To configure a Japanese user store in Oracle

1. Create a database for the user data that supports Oracle's UTF-8 NLS_CHARACTERSET encoding.
2. Using Oracle's SQL-Plus, import the smsampleusers_oracle.sql file.

   Note: More information on importing the smssampleusers_oracle.sql file exists in Sample User Directories. Be aware that if you are inserting Japanese characters, import the file from a Japanese operating system.

   The user store configuration is complete.

Appendix E: Modified Environment Variables

This section contains the following topics:

Modified Windows Environment Variables

The Policy Server installation adds and modifies the following environment variables in a Windows environment:

- NETE_PS_ROOT = $INSTALL_PATH$
- NETE_PS_PATH = $INSTALL_PATH$$/$bin; $INSTALL_PATH$$/bin/$thirdparty;$INSTALL_PATH$$/$lib
- NETEGRITY_LICENSE_FILE = %NETE_PS_ROOT%$/$license$/$license.dat
- NETE_JVM_OPTION_FILE = %NETE_PS_ROOT%$/$config$/$JVMOptions.txt
- NETE_DOC_ROOT=$INSTALL_PATH$$/$netegrity_documents
- NETE_PS_OPACK="INSTALLED"
- NETE_JRE_ROOT = HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.5 in JAVAHOME value
- NETE_JAVA_PATH=%NETE_JRE_ROOT%$/$bin;%NETE_JRE_ROOT%$/$bin$/$server
- NETE_JDK_ROOT = HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit\1.5 in JAVAHOME value
- NETE_SHORTCUTS = C:\Documents and Settings\All Users\Start Menu\Programs\SOA Security Manager

  Note: This is the default location.

Modified UNIX Environment Variables

The Policy Server installation adds and modifies the following environment variables in a UNIX environment:

- NETE_PS_ROOT = $INSTALL_PATH$
- NETE_PS_PATH = $INSTALL_PATH$$/$bin; $INSTALL_PATH$$/bin/$thirdparty;$INSTALL_PATH$$/$lib
- NETEGRITY_LICENSE_FILE = %NETE_PS_ROOT%$/$license$/$license.dat
- NETE_JVM_OPTION_FILE = %NETE_PS_ROOT%$/$config$/$JVMOptions.txt
- NETE_DOC_ROOT=$INSTALL_PATH$$/$netegrity_documents
- NETE_PS_OPACK="INSTALLED"
- NETE_JDK_ROOT = JDK_PATH
- NETE_JRE_ROOT = JRE_PATH
- NETE_JAVA_PATH=%NETE_JRE_ROOT%$/$bin;%NETE_JRE_ROOT%$/$bin$/$server