CA SiteMinder® Web Services Security

WSS Agent Guide for iPlanet Web Servers

12.51

2nd Edition

CA Technologies Product References

This document references the following CA Technologies products:

- CA SiteMinder®
- CA SiteMinder® Web Services Security (formerly CA SOA Security Manager)

Contact CA Technologies

Contact CA Support

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

- 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

Providing Feedback About Product Documentation

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

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: CA SiteMinder® Web Services Security Agent for Web Servers Introduction

This section contains the following topics:

Overview

The SiteMinder Web Services Security (WSS) Agent for Web Servers is an XML-enabled version of the CA SiteMinder Web Agent that operates with a web server to handle XML messages sent to web service implementations.

When a web consumer (client) application sends an XML message to a URL that is bound to a web service, the SiteMinder WSS Agent intercepts these messages and communicates with the Policy Server to process authentication and authorization requests before the XML message is passed on to the web service. In addition, the Policy Server can provide information that the SiteMinder WSS Agent adds to the XML message, such as a SAML assertion based on the originating client application's identity.

Note: If you have purchased CA SiteMinder®, you can also use the core Web Agent functionality of the SiteMinder WSS Agent to protect other resources on a Web server. For more information about this functionality, see the CA SiteMinder® documentation—the remainder of this chapter deals specifically with use of the SiteMinder WSS Agent to protect web services.

SiteMinder WSS Agent Functions

The SiteMinder WSS Agent performs the following tasks:

- Intercept posted XML messages to protected Web services and work with the Policy Server to determine whether or not a client application should have access.

- Ensure a client application's ability to access Web services quickly and securely. The SiteMinder WSS Agent stores contextual information about client application access privileges in a session cache. You can optimize performance by modifying the cache configuration settings.
- Support multistep and chain authentication service models by generating and consuming SAML Session Tickets and WS-Security tokens.

The SiteMinder WSS Agent and the Policy Server

To enforce web service access control, the SiteMinder WSS Agent interacts with the Policy Server, where all authentication and authorization decisions are made.

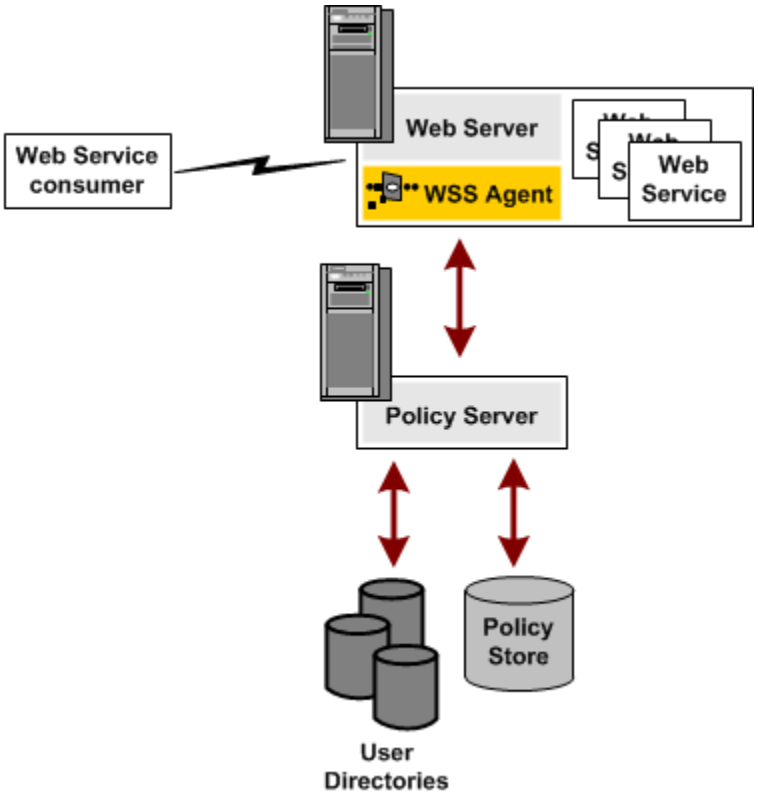
The SiteMinder WSS Agent intercepts XML messages posted to a web server and checks with the Policy Server to see if the requested resource is protected. If the resource is unprotected, the access request proceeds directly to the web server. If the resource is protected, the following occurs:

- The SiteMinder WSS Agent checks which authentication method is required for this resource. Typical credentials are a name and password, but other credentials, such as a certificate or SAML assertion, may be required.
- The SiteMinder WSS Agent obtains credentials from the transport, header, or body of the XML message.
- The SiteMinder WSS Agent passes the credentials to the Policy Server, which determines if the credentials are sufficient for the authentication method.
- If the posted XML message passes the authentication phase, the Policy Server determines if the message is authorized to access the resource. If a policy uses policy expressions as part of the authorization process, the SiteMinder WSS Agent may need to resolve the variables used in these expressions if the Policy Server cannot resolve them.
- Once the Policy Server grants access, the SiteMinder WSS Agent allows the access request to proceed to the Web service.

The SiteMinder WSS Agent can also receive message-specific attributes, in the form of *responses,* to be passed on to the Web service. A response is a personalized message or other message-specific information returned to the SiteMinder WSS Agent from the Policy Server after authorizing the message. A response consists of name-value attribute pairs that instruct the SiteMinder WSS Agent to generate SAML Session Tickets and WS-Security tokens.

SiteMinder WSS Agent Support for Web Servers

To protect Web services hosted on a web server, you deploy a SiteMinder WSS Agent on that web server (as shown in the following illustration). You then configure authentication and authorization policies for the web service resources hosted on that web server.



For a list of Web server platforms on which the SiteMinder WSS Agent is supported, see the CA SiteMinder® Web Services Security Platform Support matrix on the Technical Support site at http://ca.com/support.

Chapter 2: Preparation

This section contains the following topics:

Only iPlanet Web Server Procedures in this Guide

This guide only contains procedures for installing or configuring SiteMinder WSS Agents on iPlanet web servers.

To install or configure a SiteMinder WSS Agent on any other type of web server or operating environment, see one of the following guides:

- *SiteMinder WSS Agent for Domino Guide*.
- *SiteMinder WSS Agent for IIS Guide*
- *SiteMinder WSS Agent for Apache-based Servers Guide* .

Hardware Requirements for CA SiteMinder® Agents

Computers hosting CA SiteMinder® agents require the following hardware:

Windows operating environment requirements

agents operating on Windows operating environments require the following hardware:

- CPU: x86 or x64
- Memory: 2-GB system RAM.
- Available disk space:
  - 2-GB free disk space in the installation location.
  - .5-GB free disk space in the temporary location.

UNIX operating environment requirements

Agents operating on UNIX operating environments require the following hardware:

- CPU:
  - Solaris operating environment: SPARC
  - Red Hat operating environment: x86 or x64
- Memory: 2-GB system RAM.
- Available disk space:
  - 2-GB free disk space in the installation location.
  - .5-GB free disk space in /temp.

  Note: Daily operation of the agent requires 10 MB of free disk space in /tmp. The agent creates files and named pipes under /tmp. The path to which these files and pipes are created cannot be changed.

How to Prepare for SiteMinder WSS Agent Installation on an Oracle iPlanet Web Server

To prepare for a SiteMinder WSS Agent installation on an Oracle iPlanet server, use the following process:

1. Locate the Platform Support Matrix (see page 15). Verify that your web server supports the version of the SiteMinder WSS Agent that you want to install.
2. Verify that you have an account with one of the following types of privileges for your web server:
   - Administrative privileges (for the Windows operating environment)
   - Root privileges (for the UNIX or Linux operating environments)
3. Configure the appropriate additional settings that a SiteMinder WSS Agent requires using *one* of the following lists:
   - Oracle iPlanet web server preparations for Windows operating environments (see page 16).
   - Oracle iPlanet web server preparations for UNIX operating environments (see page 16).
   - Oracle iPlanet web server preparations for Linux operating environments (see page 17).
4. Verify that the Policy Server is installed and configured (see page 20).
5. Review the known issues section of the *CA SiteMinder® Web Services Security Release Notes (see page 22).*

Locate the Platform Support Matrix

Use the Platform Support Matrix to verify that the operating environment and other required third-party components are supported.

Follow these steps:

1. Go to the CA Support site.
2. Click Product Pages.
3. Enter the product name and click Enter.

4. Open popular links and click Informational Documentation Index.
5. Click Platform Support Matrices.

Note: You can download the latest JDK and JRE versions at the Oracle Developer Network.

Technology Partners and CA Validated Products

The latest list of partners and their validated products.

Oracle iPlanet Web Server Preparations for Windows

Oracle iPlanet servers running on Windows operating environments require the following preparations before installing a CA SiteMinder® agent:

1. For 64-bit Windows systems, verify that the Microsoft Visual C++ package prerequisite is met (see page 16).

Verify that the Microsoft Visual C++ 2005 Redistributable Package (x64) is Installed

Before installing an 12.51 CA SiteMinder® Agent on a Windows 64-bit platform, download and install the Microsoft Visual C++ 2005 Redistributable Package (x64). Go to the Microsoft downloads page, and then search for "Microsoft Visual C++ 2005 Redistributable Package (x64)."

Oracle iPlanet Web Server Preparations for UNIX

Oracle iPlanet web servers running on UNIX operating environments require the following preparations before installing a CA SiteMinder® agent:

1. Set the display variable (see page 17).
2. Verify that the appropriate patches have been installed for your operating environment:
   - Solaris patches (see page 17).
   - AIX requirements (see page 17).

Set the DISPLAY For CA SiteMinder® Agent Installations on UNIX

If you are installing the CA SiteMinder® Agent on a UNIX system from a remote terminal, such as a Telnet or Exceed terminal, be sure the DISPLAY variable is set for the local system. For example, if your machine is 111.11.1.12, set the variable as follows:

```
DISPLAY=111.11.1.12:0.0
export DISPLAY
```

Note: You can also install the agent using the console mode installation, which does not require the X window display mode.

Required Solaris Patches

Before installing a CA SiteMinder® Agent on a Solaris computer, install the following patches:

Solaris 9

Requires patch 111711-16.

Solaris 10

Requires patch 119963-08.

You can verify installed patch versions by logging in as the root user and executing the following command:

```
showrev -p | grep patch_id
```

To locate Solaris patches, go to the Oracle Solution Center.

AIX Requirements

CA SiteMinder® agents running on AIX systems require the following configurations:

- To run a rearchitected (framework) CA SiteMinder® agent for Oracle iPlanet on an AIX system, your C/C++ runtime environment must be version 8.0.0.0.

Oracle iPlanet Web Server Preparations for Linux

Oracle iPlanet web servers running on Linux operating environments require the following preparations before installing a CA SiteMinder® agent:

1. Verify that the required patches are installed (see page 18).
2. Verify that the required libraries are installed.

Required Linux Patches

The following Linux patches are required:

For Web Agents running on 64-bit Linux systems

- Binutils 2.17
- GCC 4.1.0

Required Linux Libraries

Certain library files are required for components operating on Linux operating environments. Failure to install the correct libraries can cause the following error:

```
java.lang.UnsatisfiedLinkError
```

If you are installing, configuring, or upgrading a Linux version of this component, the following packages are required on the host system:

Red Hat 5.x:

- compat–gcc-34-c++-3.4.6-*patch_version*.I386
- libstdc++-4.x.x-x.el5.i686.rpm
- libidn.so.11.rpm
- ncurses

Red Hat 6.x:

- libstdc++-4.x.x-x.el6.i686.rpm
- libidn-1.18-2.el6.i686
- libXext.i686.rpm
- libXrender.i686.rpm
- linXtst.i686.rpm
- libidn.so.11.rpm
- ncurses

Additionally, for Red Hat 6.x (64-bit):

All the RPM packages that are required for 64-bit Red Hat 6.x are *32-bit* packages.

- libXau-1.0.5-1.el6.i686.rpm
- libxcb-1.5-1.el6.i686.rpm
- compat-db42-4.2.52-15.el6.i686.rpm
- compat-db43-4.3.29-15.el6.i686.rpm
- libX11-1.3-2.el6.i686.rpm
- libXrender-0.9.5-1.el6.i686.rpm
- libexpat.so.1 (provided by expat-2.0.1-11.el6_2.i686.rpm)
- libfreetype.so.6 (provided by freetype-2.3.11-6.el6_2.9.i686.rpm)
- libfontconfig.so.1 (provided by fontconfig-2.8.0-3.el6.i686.rpm)
- libICE-1.0.6-1.el6.i686.rpm
- libuuid-2.17.2-12.7.el6.i686.rpm
- libSM-1.1.0-7.1.el6.i686.rpm
- libXext-1.1-3.el6.i686.rpm
- compat-libstdc++-33-3.2.3-69.el6.i686.rpm
- compat-db-4.6.21-15.el6.i686.rpm
- libXi-1.3-3.el6.i686.rpm
- libXtst-1.0.99.2-3.el6.i686.rpm
- libXft-2.1.13-4.1.el6.i686.rpm
- libXt-1.0.7-1.el6.i686.rpm
- libXp-1.0.0-15.1.el6.i686.rpm
- libstdc++.i686.rpm
- compat-libtermcap.rpm

- libidn.i686.rpm
- ncurses

Policy Server Requirements

Verify the following criteria:

- Your Policy Server is installed and configured.
- Your Policy server can communicate with the computer where you plan to install the agent.

To install and configure a CA SiteMinder® agent, a Policy Server requires at least the following items:

- A CA SiteMinder® administrator that has the right to register trusted hosts.

  A trusted host is a client computer where one or more CA SiteMinder® Agents are installed and registered with the Policy Server. The CA SiteMinder® administrator must have permissions to register trusted hosts with the Policy Server. Registering a trusted host creates a unique trusted host name object on the Policy Server.

- An Agent identity

  An Agent identity establishes a mapping between the Policy Server and the name or IP address of the web server instance hosting an Agent. You define an Agent identity from the Agents object in the Administrative UI. You assign it a name and specify the Agent type as a Web Agent.

- A Host Configuration Object (HCO)

  The host configuration object on the Policy Server defines the communication between the agent and the Policy Server that occurs after an initial connection. The Initial connections use the parameters in the SmHost.conf file.

- Agent Configuration Object (ACO)

  This object includes the parameters that define the agent configuration. All CA SiteMinder® agents require at least one of the following configuration parameters that are defined in the ACO:

AgentName

Defines the identity of the web agent. This identity links the name and the IP address or FQDN of each web server instance hosting an Agent.

The value of the DefaultAgentName is used instead of the AgentName parameter if any of the following events occur:

- The AgentName parameter is disabled.
- The value of AgentName parameter is empty.
- The values of the AgentName parameter do *not* match any existing agent object.

Note: This parameter can have more than one value. Use the multivalue option when setting this parameter in an Agent Configuration Object. For local configuration files, add each value to a separate line in the file.

Default: No default

Limit: Multiple values are allowed, but each AgentName parameter has a 4,000 character limit. Create additional AgentName parameters as needed by adding a character to the parameter name. For example, AgentName, AgentName1, AgentName2.

Limits: Must contain 7-bit ASCII characters in the range of 32-127, and include one or more printable characters. Cannot contain the ampersand (&) and asterisk (*) characters. The value is not case-sensitive. For example, the names MyAgent and myagent are treated the same.

Example: myagent1,192.168.0.0 (IPV4)

Example: myagent2, 2001:DB8::/32 (IPV6)

Example: myagent,www.example.com

Example (multiple AgentName parameters): AgentName1, AgentName2, AgentName3. The value of each AgentName*number* parameter is limited to 4,000 characters.

DefaultAgentName

Defines a name that the agent uses to process requests. The value for DefaultAgentName is used for requests on an IP address or interface when no agent name value exists in the AgentName parameter.

If you are using virtual servers, you can set up your CA SiteMinder® environment quickly by using a DefaultAgentName. Using DefaultAgentName means that you do not need to define a separate agent for each virtual server.

Important! If you do not specify a value for the DefaultAgentName parameter, then the value of the AgentName parameter requires every agent identity in its list. Otherwise, the Policy Server cannot tie policies to the agent.

Default: No default.

Limit: Multiple values are allowed.

Limits: Must contain 7-bit ASCII characters in the range of 32-127, and include one or more printable characters. Cannot contain the ampersand (&) and asterisk (*) characters. The value is not case-sensitive. For example, the names MyAgent and myagent are treated the same.

Review the CA SiteMinder® Web Services Security Release Notes for Known Issues

The most-recent versions of the CA SiteMinder® Web Services Security Release notes are available from the CA Support website. We recommend reviewing them before installing or configuring a SiteMinder WSS Agent.

Follow these steps:

1. Open a web browser and navigate to the Technical Support website.
2. Click Enterprise/Small and Medium Business.

   The Support for Businesses and Partners page appears.
3. Under the Get Support tab, click Product Documentation.

   The documentation page appears.
4. Click the field under Select a Bookshelf.
5. Type siteminder.

   A list of CA SiteMinder® bookshelves appears.
6. Click the bookshelf that you want from the list, and then click Go.

   The bookshelf opens (in a new window or tab, depending on your browser settings).
7. Click Release Notes.

   A list of release notes appears.
8. Click *one* of the following links to display the Release Notes in format you want:
   - View HTML
   - Download PDF

Chapter 3: Install and Configure SiteMinder WSS Agents for iPlanet on Windows

This section contains the following topics:

Agent Installation Compared to Agent Configuration

The concepts of installation and configuration have specific meanings when used to describe CA SiteMinder® agents.

Installation means installing the CA SiteMinder® agent software on a computer system. For example, installing an agent creates directories and copies the CA SiteMinder® agent software and other settings to the computer.

Configuration occurs after installation and means the act of preparing the CA SiteMinder® agent software for a specific web server on a computer. This preparation includes registering the agent with CA SiteMinder® Policy Servers, and creating a runtime server instance for the web server that is installed on the computer.

Use the wizard-based installation and configuration programs to install and configure your agent on your first web server. The wizard-based programs create a .properties file.

Use the .properties file and the respective executable file to install or configure the agent silently on additional web servers.

Set the JRE in the Path Variable

Set the Java Runtime Environment (JRE) in the Windows path variable.

Follow these steps:

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

Apply the Unlimited Cryptography Patch to the JRE

Patch the Java Runtime Environment (JRE) used by the Agent to support unlimited key strength in the Java Cryptography Extension (JCE) package. The patches for all supported platforms are available from the Oracle website.

The files that need to be patched are:

- local_policy.jar
- US_export_policy.jar

The local_policy.jar and US_export_policy.jar files can found be in the following locations:

- Windows

  jre_home\lib\security

- UNIX

  jre_home/lib/security

jre_home

Defines the location of your Java Runtime Environment installation.

Configure the JVM to Use the JSafeJCE Security Provider

The SiteMinder WSS Agent XML encryption function requires that the JVM is configured to use the JSafeJCE security provider.

Follow these steps:

1. Add a security provider entry for JSafeJCE (com.rsa.jsafe.provider.JsafeJCE) to the java.security file located in the following location:

   - *JRE_HOME*\lib\security (Windows)
   - *JRE_HOME*/lib/security (UNIX)

   JRE_HOME

   Is the installed location of the JRE used by the application server.

   In the following example, the JSafeJCE security provider entry has been added as the second security provider:

   ```
   security.provider.1=sun.security.provider.Sun
   security.provider.2=com.rsa.jsafe.provider.JsafeJCE
   security.provider.3=sun.security.rsa.SunRsaSign
   security.provider.4=com.sun.net.ssl.internal.ssl.Provider
   security.provider.5=com.sun.crypto.provider.SunJCE
   security.provider.6=sun.security.jgss.SunProvider
   security.provider.7=com.sun.security.sasl.Provider
   ```

   Note: If using the IBM JRE, always configure the JSafeJCE security provider immediately after (that is with a security provider number one higher than) the IBMJCE security provider (com.ibm.crypto.provider.IBMJCE)

2. Add the following line to *JRE_HOME*\lib\security\java.security (Windows) or *JRE_HOME*/lib/security/java.security (UNIX) to set the *initial* FIPS mode of the JsafeJCE security provider:

   ```
   com.rsa.cryptoj.fips140initialmode=NON_FIPS140_MODE
   ```

   Note: The initial FIPS mode does not affect the final FIPS mode you select for the SiteMinder WSS Agent.

How to Install and Configure a SiteMinder WSS Agent for iPlanet on a Windows System

Installing CA SiteMinder® agents on the Windows operating environment requires several separate procedures. To install and configure an SiteMinder WSS Agent on Windows, use the following process:

1. Gather the required information for the installation program (see page 26).
2. Gather the required information for the configuration program (see page 26).
3. Run the CA SiteMinder® Web Services Security installation program (see page 29).
4. Run the configuration program (see page 29).
5. (Optional) Install and configure additional <agents> silently (see page 30).

Gather the Information for the Installation Program

Gather the following information about your web server before running the installation program for the agent:

Installation Directory

Specifies the location of the agent binary files on your web server. The *web_agent_home* variable is set to this location.

Limit: The product requires the name "webagent" for the bottom directory in the path

Gather Information Required for SiteMinder WSS Agent Configuration

The following information must be supplied during Trusted Host registration:

SM Admin User Name

The name of a Policy Server administrator allowed to register the host with the Policy Server.

This administrator should already be defined at the Policy Server and have the permission Register Trusted Hosts set. The default administrator is SiteMinder.

SM Admin Password

The Policy Server administrator account password.

Trusted Host Name

Specifies a unique name that represents the trusted host to the Policy Server. This name *does not* have to be the same as the physical client system that you are registering; it can be any unique name, for example, mytrustedhost.

Note: This name must be unique among trusted hosts and not match the name of any other Agent.

Host Configuration Object

The name of the Host Configuration Object in the Policy Server that defines the connection between the trusted host and the Policy Server. For example, to use the default, enter DefaultHostSettings. In most cases, you will have created your own Host Configuration Object.

Note: This value must match the Host Configuration Object entry preconfigured on the Policy Server.

Policy Server IP Address

The IP address, or host name, and authentication port of the Policy Server where you are registering the host. The default port is 44442. If you do not provide a port, the default is used.

You can specify a non-default port number, but if your Policy Server is configured to use a non-default port and you omit it when you register a trusted host, the following error is displayed:

Registration Failed (bad ipAddress[:port] or unable to connect to Authentication server (-1)

Note also that if you specify a non-default port, that port is used for the Policy Server's authentication, authorization, and accounting ports; however, the unified server responds to any Agent request on any port. The entry in the SmHost.conf file will look like:

policyserver="*ip_address*,5555,5555,5555"

FIPS Encryption Mode

Determines whether the Agent communicates with the Policy Server using certified Federal Information Processing Standard (FIPS) 140-2 compliant cryptographic libraries.

FIPS Compatibility Mode (Default)

Specifies non-FIPS mode, which lets the Policy Server and the Agents read and write information using the existing CA SiteMinder® encryption algorithms. If your organization does not require the use of FIPS-compliant algorithms, the Policy Server and the Agents can operate in non-FIPS mode without further configuration.

FIPS Only Mode

Specifies full-FIPS mode, which requires that the Policy Server and Web Agents read and write information using only FIPS 140-2 algorithms.

Important! A CA SiteMinder® installation that is running in Full FIPS mode cannot interoperate with, or be backward compatible to, earlier versions of CA SiteMinder®, including all agents, custom software using older versions of the Agent API, and custom software using PM APIs or any other API that the Policy Server exposes. You must re-link all such software with the corresponding versions of the respective SDKs to achieve the required support for Full FIPS mode.

Run the Installer to Install a SiteMinder WSS Agent

Install the SiteMinder WSS Agent using the CA SiteMinder® Web Services Security installation media on the Technical Support site.

Follow these steps:

1. Exit all applications that are running.
2. Navigate to the installation material.
3. Double-click ca-sm-wss-12.51-*cr*-win32.exe.

   cr

   Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

   The CA SiteMinder® Web Services Security installation wizard starts.

   Important! If you are running this wizard on Windows Server 2008, run the executable file with Administrator permissions, even if you are logged into the system as an Administrator.

4. Use gathered system and component information to install the SiteMinder WSS Agent. Consider the following points when running the installer:

   - When prompted to select which CA SiteMinder® Web Services Security Agents to install, select **CA SiteMinder® Web Services Security Agent for Web Servers**.
   - When prompted to select the Java version, the installer lists all Java executables present on the system. Select a supported 32-bit Java Runtime Environment (refer to the Platform Support Matrix on the Technical Support site).
   - If you enter path information in the wizard by cutting and pasting, enter (and delete, if necessary) at least one character to enable the Next button.
   - If the installer detects the presence of an existing CA SiteMinder® Web Agent, it displays a warning dialog stating that the install will upgrade the Web Agent. Click Continue to upgrade the Web Agent to a SiteMinder WSS Agent. If you proceed, the software upgrade occurs in the installed location of the existing Web Agent.

5. Review the information that is presented on the Pre-Installation Summary page, then click Install.

   Note: If the installation program detects that newer versions of certain system DLLs are installed on your system, it asks if you want to overwrite these newer files with older files. Select No To All if you see this message.

   The SiteMinder WSS Agent files are copied to the specified location.

6. On the CA SiteMinder® Web Services Security Configuration screen, click one of the following options and click Next:

   - Yes. I would like to configure CA SiteMinder® Web Services Security Agents now.
   - No. I will configure CA SiteMinder® Web Services Security Agents later.

   If the installation program detects that there are locked Agent files, it prompts you to restart your system instead of reconfiguring it. Select whether to restart the system automatically or later on your own.

7. Click Done.

   If you selected the option to configure SiteMinder WSS Agents now, the installation program prepares the CA SiteMinder® Web Services Security Configuration Wizard and begins the trusted host registration and configuration process. Use the information that you gathered earlier to complete the wizard.

   If you did not select the option to configure SiteMinder WSS Agents now, or if you are required to reboot the system after installation, run the configuration wizard manually later.

Installation Notes:

- After installation, you can review the installation log file in *WSS_HOME*\install_config_info. The file name is: CA_SiteMinder_Web_Services_Security_Install_*install-date-and-time*.log

  WSS_Home

  Specifies the path to where CA SiteMinder® Web Services Security is installed.

  Default: C:\Program Files\CA\Web Services Security

  install-date-and-time

  Specifies the date and time that the SiteMinder WSS Agent was installed.

- The Agent cannot communicate properly with the Policy Server until the trusted host is registered.

Run the SiteMinder WSS Agent Configuration Program on Windows

After gathering the information for your agent configuration, run the agent configuration program. This program creates an agent runtime instance for the web servers running on your computer.

This configuration program is wizard or console based, depending on the option you select. Running the configuration program in the wizard or console mode once creates a properties file. Use the properties file to run unattended configurations on other computers with same operating environment in the future.

Follow these steps:

1. Open the following directory on your web server:

   WSS_Home\install_config_info

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.

   Default: C:\Program Files\CA\Web Services Security

2. Use *one* of the following configuration methods:

   - For a GUI-based configuration, right-click ca-pep-config.exe, and then select Run as Administrator:
   - For a console-based configuration, enter the following command from a Command Prompt window with Administrator privileges open to *WSS_Home*\install_config_info:

     ```
     ca-pep-config.exe -i console
     ```

     Important! If you are running this wizard on Windows Server 2008, run the executable file with administrator permissions. Use these permissions even if you are logged in to the system as an administrator. For more information, see the release notes for your CA SiteMinder® component.

3. Use the information you gathered earlier to complete the wizard.

   The agent runtime instance is created for your web servers.

(Optional) Run the Unattended or Silent Installation and Configuration Programs Subsequent SiteMinder WSS Agents on Windows

The unattended or silent installation option can help you automate the installation and configuration process. This method saves time if you have a large CA SiteMinder® Web Services Security environment that uses many agents with identical settings.

For example, suppose the Agents in your environment use the same web server version, installation directory, Agent Configuration Object and Policy Servers. Use the installation wizard or console-based installation program for your first installation. Afterwards, you could create your own script to run the installation program with the .properties file the wizard or console-based installation program created.

Follow these steps:

1. Run the following wizards on your first web server (in the order shown):
   a. The CA SiteMinder® Web Services Security Installation wizard.
   b. The CA SiteMinder® Web Services Security Configuration wizard.
2. Locate the following file on your first web server:

   WSS_Home\install_config_info\ca-wss-installer.properties

   Note: If the path contains spaces, surround it with quotes.

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.

   Default: C:\Program Files\CA\Web Services Security
3. Perform each of the following steps on the other web servers in your environment:

   Note: To automate this process, create your own customized script to execute these files on your systems. Use any scripting language that you want.
   a. Create a temporary directory on the subsequent web server.
   b. Copy the following files from your first web server (from Steps 1 and 2) to the temporary directory on your subsequent web server:
      - The SiteMinder WSS Agent Installation executable file.
      - The ca-pepconfig-installer.properties file.
   c. Open a Command Prompt window with Administrative privileges in the temporary directory.
   d. Run the following command:

      ca-sm-wss-12.51-*cr*-win32.exe -f *properties_file* -i silent.

      cr

      Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

      Important! If you are running this wizard on Windows Server 2008, run the executable file with Administrator permissions, even if you are logged into the system as an Administrator.

      The SiteMinder WSS Agent is installed and configured on the subsequent server automatically.
   e. (Optional) Delete the temporary directory from your subsequent web server.
4. Repeat Step 3 for each additional web server in your CA SiteMinder® environment that uses the configuration that the settings in your ca-wss-installer.properties file specify.

Apply CA SiteMinder® Changes to Oracle iPlanet Configuration Files with Oracle iPlanet Administration Server Console for SunOne 6.1 Servers

The Agent Configuration Wizard modifies the default obj.conf, and mime.types files that the Oracle iPlanet web server uses.

If you are using version 6.1 of a SunOne web server, and you plan to use the Oracle iPlanet Administration console, apply the changes to these files *before* using the console. If you do not apply the changes using the console first, the changes that are made for your CA SiteMinder® configuration could be corrupted. If you lose your configuration, run the configuration program again.

Note: The agent adds settings to the obj.conf file of the Oracle iPlanet web server when the Agent is configured to support an advanced authentication scheme. CA SiteMinder® does *not* remove these settings later. Edit the obj.conf file manually to remove any obsolete settings.

Follow these steps:

1. Log in to the Oracle iPlanet Administration Server console.
2. From the Servers tab, select the web server with the CA SiteMinder® agent installed and click Manage.
3. In the right corner of the dialog, click Apply.

   A warning message about loading the modified configuration files appears.
4. Click Load Configuration Files.
5. Exit the console.
6. Restart the web server.
7. Optimize the Agent for Oracle iPlanet by tuning the shared memory segments.

   The CA SiteMinder® changes are applied.

More Information

Reconfigured Web Agent Won't Operate (see page 114)

Manually Configure Non-Default Server Instances, Virtual Servers, or Reverse Proxies for Oracle iPlanet Web Servers

The SiteMinder WSS Agent Configuration wizard only configures the default instance of your Oracle iPlanet web server. To configure a different instance of the Oracle iPlanet web server for CA SiteMinder®, manually edit the obj.conf file that is associated with that server instance. Examples of server instances that need manual configuration include:

- Servers installed in a nondefault directory
- Servers that you want to configure as a reverse proxy. We recommend configuring the reverse proxy using your Oracle iPlanet interface *before* adding the CA SiteMinder® settings to the obj.conf file.

  Note: The CA SiteMinder® Agent Configuration wizard *only* modifies the *default* obj.conf file on the Oracle iPlanet (formerly Sun Java System) web server. To protect other instances or reverse proxy deployments with CA SiteMinder®, copy the CA SiteMinder® settings from the default obj.conf file to any respective *instance_name*-obj.conf files. For example, your web server created an obj.conf file when you installed it, but you later added a server instance named my_server.example.com. To protect resources on my_server.example.com with CA SiteMinder®, copy the CA SiteMinder® settings the wizard added from the obj.conf file to the my_server.example.com-obj.conf file.
- Virtual servers on the same computer

Note: SunOne/Sun Java 7.0 web servers do *not* require these manual configuration steps.

Follow these steps:

1. Locate the directory of the server instance you want to configure.
2. Open the obj.conf file with a text editor.
3. Locate the following line:

   ```
   <Object name="default">
   ```

4. Insert a new line below the previous one, and then add the following text:

   ```
   AuthTrans fn="SiteMinderAgent"
   ```

5. Locate the following line:

   ```
   AuthTrans fn="match-browser" browser="*MSIE*" ssl-unclean-shutdown="true"
   ```

6. Insert a new line below the previous one, and then add the following text:

   ```
   NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi" dir="agent_home/pw"
   name="cgi"
   NameTrans fn="pfx2dir" from="/siteminderagent/pw" dir="agent_home/pw"
   NameTrans fn="pfx2dir" from="/siteminderagent/jpw" dir="agent_home/jpw"
   ```

```
NameTrans fn="pfx2dir" from="/siteminderagent/redirectjsp"
dir="agent_home/affwebservices/redirectjsp"
NameTrans fn="pfx2dir" from="/siteminderagent/certoptional"
dir="agent_home/samples"
NameTrans fn="pfx2dir" from="/siteminderagent" dir="agent_home/samples"
NameTrans fn="pfx2dir" from="/siteminderagent/pwservlet" dir=agent_home/jpw"
```

agent_home

Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

7. Locate the following line:

   ```
   NameTrans fn="ntrans-j2ee" name="j2ee"
   ```

8. Insert a new line below the previous one, and then add the following text:

   ```
   PathCheck fn="SmRequireAuth"
   ```

9. Remove the following line:

   ```
   NameTrans fn="pfx2dir" from="/mc-icons" dir="C:/Program
   Files/Sun/WebServer7.0/lib/icons" name="es-internal"
   ```

10. Locate the following line:

    ```
    ObjectType fn="force-type" type="text/plain"
    ```

11. Insert a new line below the previous one, and then add the following text:

    ```
    Service method="(GET|POST)" fn="SmAdvancedAuth"
    ```

12. Locate the following line:

    ```
    Error fn="error-j2ee
    ```

13. Insert a new line above the previous one, and then add the following text:

    ```
    Error fn="SmSoapFault" code="500" reason="SmSoapFault"
    ```

14. Save the obj.conf file.
15. Open the magnus.conf file with a text editor.
16. Add the following line:

    ```
    Init fn="load-modules" shlib="agent_home/bin/SunOneWebAgent.dll"
    funcs="SmInitAgent,SmInitChild,SiteMinderAgent,SmRequireAuth,SmAdvancedAuth,S
    mSoapFault
    ```

17. Save the magnus.conf file.

    The Oracle iPlanet web server is manually configured.

(Optional) Improve Server Performance with httpd.conf File Changes

You can improve server performance by modifying the default configuration settings in the httpd.conf file; however, these changes are *not* required:

Follow these steps:

1. For Oracle iPlanet web servers, assign a higher priority level to your Apache20WebAgent.dll file than any other auth modules or access modules on your web server.
2. For low-traffic websites, define the following directives:
   - Set MaxRequestsPerChild>1000 *or* Set MaxRequestsPerChild=0
   - MinSpareServers >5
   - MaxSpareServers>10
   - StartServers=MinSpareServers>5

3. For high-traffic websites, define the following directives:
   - Set MaxRequestsPerChild>3000 *or* Set MaxRequestsPerChild=0
   - MinSpareServers >10
   - MaxSpareServers>15
   - StartServers=MinSpareServers>10

Chapter 4: Install and Configure SiteMinder WSS Agents for iPlanet on UNIX/Linux

This section contains the following topics:

Agent Installation Compared to Agent Configuration

The concepts of installation and configuration have specific meanings when used to describe CA SiteMinder® agents.

Installation means installing the CA SiteMinder® agent software on a computer system. For example, installing an agent creates directories and copies the CA SiteMinder® agent software and other settings to the computer.

Configuration occurs after installation and means the act of preparing the CA SiteMinder® agent software for a specific web server on a computer. This preparation includes registering the agent with CA SiteMinder® Policy Servers, and creating a runtime server instance for the web server that is installed on the computer.

Use the wizard-based installation and configuration programs to install and configure your agent on your first web server. The wizard-based programs create a .properties file.

Use the .properties file and the respective executable file to install or configure the agent silently on additional web servers.

Set the JRE in the PATH Variable

Set the Java Runtime Environment (JRE) in the UNIX system PATH variable.

To set the JRE in the PATH variable

1. Open a Command Window.
2. Run the following commands:

```
PATH=$PATH:JRE_HOME
export PATH
```

JRE_HOME

Defines the installed location of your Java Runtime Environment.

Apply the Unlimited Cryptography Patch to the JRE

Patch the Java Runtime Environment (JRE) used by the Agent to support unlimited key strength in the Java Cryptography Extension (JCE) package. The patches for all supported platforms are available from the Oracle website.

The files that need to be patched are:

- local_policy.jar
- US_export_policy.jar

The local_policy.jar and US_export_policy.jar files can found be in the following locations:

- Windows

  jre_home\lib\security

- UNIX

  jre_home/lib/security

jre_home

Defines the location of your Java Runtime Environment installation.

Configure the JVM to Use the JSafeJCE Security Provider

The SiteMinder WSS Agent XML encryption function requires that the JVM is configured to use the JSafeJCE security provider.

Follow these steps:

1. Add a security provider entry for JSafeJCE (com.rsa.jsafe.provider.JsafeJCE) to the java.security file located in the following location:

   - *JRE_HOME*\lib\security (Windows)
   - *JRE_HOME*/lib/security (UNIX)

   JRE_HOME

   Is the installed location of the JRE used by the application server.

   In the following example, the JSafeJCE security provider entry has been added as the second security provider:

   ```
   security.provider.1=sun.security.provider.Sun
   security.provider.2=com.rsa.jsafe.provider.JsafeJCE
   security.provider.3=sun.security.rsa.SunRsaSign
   security.provider.4=com.sun.net.ssl.internal.ssl.Provider
   security.provider.5=com.sun.crypto.provider.SunJCE
   security.provider.6=sun.security.jgss.SunProvider
   security.provider.7=com.sun.security.sasl.Provider
   ```

   Note: If using the IBM JRE, always configure the JSafeJCE security provider immediately after (that is with a security provider number one higher than) the IBMJCE security provider (com.ibm.crypto.provider.IBMJCE)

2. Add the following line to *JRE_HOME*\lib\security\java.security (Windows) or *JRE_HOME*/lib/security/java.security (UNIX) to set the *initial* FIPS mode of the JsafeJCE security provider:

   ```
   com.rsa.cryptoj.fips140initialmode=NON_FIPS140_MODE
   ```

   Note: The initial FIPS mode does not affect the final FIPS mode you select for the SiteMinder WSS Agent.

How to Install SiteMinder WSS Agents for Web Servers on UNIX or Linux Systems

Installing CA SiteMinder® agents on the UNIX or Linux operating environments requires several separate procedures. These procedures are described using the following process:

1. Gather the information that the installation program requires (see page 40).
2. Do one of the following procedures:
   - Run the installer to install a SiteMinder WSS Agent using a GUI (see page 44).
   - Run the installer to install a SiteMinder WSS Agent using a UNIX console (see page 42).

Gather the Information for the Installation

Gather the following information about your web server before running the installation program for the agent:

Installation Directory

Specifies the location of the agent binary files on your web server. The *web_agent_home* variable is set to this location.

Limit: The product requires the name webagent for the bottom directory in the path.

Gather Information Required for SiteMinder WSS Agent Configuration

The following information must be supplied during Trusted Host registration:

SM Admin User Name

The name of a Policy Server administrator allowed to register the host with the Policy Server.

This administrator should already be defined at the Policy Server and have the permission Register Trusted Hosts set. The default administrator is SiteMinder.

SM Admin Password

The Policy Server administrator account password.

Trusted Host Name

Specifies a unique name that represents the trusted host to the Policy Server. This name *does not* have to be the same as the physical client system that you are registering; it can be any unique name, for example, mytrustedhost.

Note: This name must be unique among trusted hosts and not match the name of any other Agent.

Host Configuration Object

The name of the Host Configuration Object in the Policy Server that defines the connection between the trusted host and the Policy Server. For example, to use the default, enter DefaultHostSettings. In most cases, you will have created your own Host Configuration Object.

Note: This value must match the Host Configuration Object entry preconfigured on the Policy Server.

Policy Server IP Address

The IP address, or host name, and authentication port of the Policy Server where you are registering the host. The default port is 44442. If you do not provide a port, the default is used.

You can specify a non-default port number, but if your Policy Server is configured to use a non-default port and you omit it when you register a trusted host, the following error is displayed:

Registration Failed (bad ipAddress[:port] or unable to connect to Authentication server (-1)

Note also that if you specify a non-default port, that port is used for the Policy Server's authentication, authorization, and accounting ports; however, the unified server responds to any Agent request on any port. The entry in the SmHost.conf file will look like:

policyserver="*ip_address*,5555,5555,5555"

FIPS Encryption Mode

Determines whether the Agent communicates with the Policy Server using certified Federal Information Processing Standard (FIPS) 140-2 compliant cryptographic libraries.

FIPS Compatibility Mode (Default)

Specifies non-FIPS mode, which lets the Policy Server and the Agents read and write information using the existing CA SiteMinder® encryption algorithms. If your organization does not require the use of FIPS-compliant algorithms, the Policy Server and the Agents can operate in non-FIPS mode without further configuration.

FIPS Only Mode

Specifies full-FIPS mode, which requires that the Policy Server and Web Agents read and write information using only FIPS 140-2 algorithms.

Important! A CA SiteMinder® installation that is running in Full FIPS mode cannot interoperate with, or be backward compatible to, earlier versions of CA SiteMinder®, including all agents, custom software using older versions of the Agent API, and custom software using PM APIs or any other API that the Policy Server exposes. You must re-link all such software with the corresponding versions of the respective SDKs to achieve the required support for Full FIPS mode.

Run the Installer to Install a SiteMinder WSS Agent Using a UNIX Console

Install the SiteMinder WSS Agent using the CA SiteMinder® Web Services Security installation media on the Technical Support site. Consider the following:

- Depending on your permissions, you may need to add executable permissions to the install file by running the following command:

  `chmod +x ca-sm-wss-12.51-`*`cr-unix_version`*`.bin`

  cr

  Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

  unix_version

  Specifies the UNIX version: **sol** or **linux**.

- If you execute the CA SiteMinder® Web Services Security installer across different subnets, it can crash. Install CA SiteMinder® Web Services Security components directly on the host system to avoid the problem.

To install the SiteMinder WSS Agent

1. Exit all applications that are running.
2. Open a shell and navigate to where the install program is located.
3. Enter the following command:

   `./ca-sm-wss-12.51-`*`cr`*`-unix_version.bin -i console`

   The CA SiteMinder® Web Services Security installer starts.
4. Use gathered system and component information to install the SiteMinder WSS Agent. Consider the following as you make your selections:
   - When prompted to select what agents to install, select **CA SiteMinder® Web Services Security Agent for Web Servers**.
   - When prompted to select the Java version, the installer lists all Java executables present on the system. Select a supported 32-bit Java Runtime Environment (refer to the Platform Support Matrix on the Technical Support site).
   - Do not uses spaces in the SiteMinder WSS Agent install path.
   - If the installer detects the presence of an existing CA SiteMinder® Web Agent, it displays a warning dialog stating that the install will upgrade the Web Agent. Click Continue to upgrade the Web Agent to a SiteMinder WSS Agent. If you proceed, the software upgrade occurs in the installed location of the existing Web Agent.

5. Review the information presented on the Pre-Installation Summary page, then proceed.

   Note: If the installation program detects that newer versions of certain system libraries are installed on your system it asks if you want to overwrite these newer files with older files. Select No To All if you see this message.

   The SiteMinder WSS Agent files are copied to the specified location. Afterward, the CA CA SiteMinder® Web Services Security Configuration screen is displayed.

6. Select one of the following options:

   - Yes. I would like to configure CA SiteMinder® Web Services Security Agents now.
   - No. I will configure CA SiteMinder® Web Services Security Agents later.

7. Hit Enter.

   If you selected the option to configure SiteMinder WSS Agents now, the installation program prepares the CA SiteMinder® Web Services Security Configuration Wizard and begins the trusted host registration and configuration process.

   If you did not select the option to configure SiteMinder WSS Agents now or if you are required to reboot the system after installation you must start the configuration wizard manually later.

Installation Notes:

- To check if the unattended installation completed successfully, see the CA_SiteMinder_Web_Services_Security_Install_*install-date-and-time*.log file in *WSS_HOME/*install_config_info directory. This log file contains the results of the installation.

  WSS_Home

  Specifies the path to where CA SiteMinder® Web Services Security is installed.

  install-date-and-time

  Specifies the date and time that the SiteMinder WSS Agent was installed.

- The Agent cannot communicate properly with the Policy Server until the trusted host is registered.

- The Agent cannot communicate properly with the Policy Server until the trusted host is registered.

Run the Installer to Install a SiteMinder WSS Agent Using a GUI

Install the SiteMinder WSS Agent using the CA SiteMinder® Web Services Security installation media on the Technical Support site. Consider the following:

- Depending on your permissions, you may need to add executable permissions to the install file by running the following command:

  `chmod +x ca-sm-wss-12.51-`*`cr-unix_version`*`.bin`

  cr

  Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

  unix_version

  Specifies the UNIX version: **sol** or **linux**.

- If you execute the CA SiteMinder® Web Services Security installer across different subnets, it can crash. Install CA SiteMinder® Web Services Security components directly on the host system to avoid the problem.

To install the SiteMinder WSS Agent

1. Exit all applications that are running.
2. Open a shell and navigate to where the install program is located.
3. Enter the following command:

   `./ca-sm-wss-12.51-`*`cr-unix_version`*`.bin`

   The CA SiteMinder® Web Services Security installer starts.
4. Use gathered system and component information to install the SiteMinder WSS Agent. Consider the following when running the installer:
   - When prompted to select what agents to install, select **CA SiteMinder® Web Services Security Agent for Web Servers**
   - When prompted to select the Java version, the installer lists all Java executables present on the system. Select a supported 32-bit Java Runtime Environment (refer to the Platform Support Matrix on the Technical Support site).
   - If you enter path information in the wizard by cutting and pasting, enter (and delete, if necessary) at least one character to enable the Next button.
   - If the installer detects the presence of an existing CA SiteMinder® Web Agent, it displays a warning dialog stating that the install will upgrade the Web Agent. Click Continue to upgrade the Web Agent to a SiteMinder WSS Agent. If you proceed, the software upgrade occurs in the installed location of the existing Web Agent.
   - Do *not* use spaces in the SiteMinder WSS Agent install path.

5. Review the information presented on the Pre-Installation Summary page, then click Install.

   Note: If the installation program detects that newer versions of certain system libraries are installed on your system it asks if you want to overwrite these newer files with older files. Select No To All if you see this message.

   The SiteMinder WSS Agent files are copied to the specified location. Afterward, the CA CA SiteMinder® Web Services Security Configuration screen is displayed.

6. Select one of the following options:

   - Yes. I would like to configure CA SiteMinder® Web Services Security Agents now.
   - No. I will configure CA SiteMinder® Web Services Security Agents later.

7. Click Done.

   If you selected the option to configure SiteMinder WSS Agents now, the installation program prepares the CA SiteMinder® Web Services Security Configuration Wizard and begins the trusted host registration and configuration process.

   If you did not select the option to configure SiteMinder WSS Agents now or if you are required to reboot the system after installation you must start the configuration wizard manually later.

Installation Notes:

- To check if the unattended installation completed successfully, see the CA_SiteMinder_Web_Services_Security_Install_*install-date-and-time*.log file in *WSS_HOME/*install_config_info directory. This log file contains the results of the installation.

  WSS_Home

  Specifies the path to where CA SiteMinder® Web Services Security is installed.

  install-date-and-time

  Specifies the date and time that the SiteMinder WSS Agent was installed.

- The Agent cannot communicate properly with the Policy Server until the trusted host is registered.

How to Configure SiteMinder WSS Agents on UNIX/Linux

Configuring the SiteMinder WSS Agent occurs after the installation. Configuration requires several separate procedures which are described using the following process:.

1. Set environment variables (see page 46).
2. Run the agent configuration program. (see page 47)
3. (Optional) Run the unattended or silent installation and configuration program for other agents (see page 48).
4. Determine if your Agent for Oracle iPlanet requires any of the following additional configuration steps:
   - (For SunOne 6.1 web servers only) If you want to use the Oracle iPlanet Administration Server console, apply the CA SiteMinder® changes to the configuration files of the Oracle iPlanet web server (see page 49).
   - (Except SunOne 7.0/Sun Java 7.0 web servers) Manually configure any nondefault server instances, reverse proxies, or virtual servers for CA SiteMinder® (see page 33).
   - For Solaris 9 SP3 and Solaris 10, modify the startup script (see page 53).

Set Environment Variables for a SiteMinder WSS Agent on UNIX

After installing the SiteMinder WSS Agent on UNIX, you must set required environment variables using the ca_wa_env.sh script. Running the script for SiteMinder WSS Agents on most UNIX platforms ensures that the SiteMinder WSS Agent and web server can work together.

The ca_wa_env.sh script sets the following environment variables:

- NETE_WA_ROOT
- PATH
- NETE_WA_PATH
- LD_LIBRARY_PATH

  Note: The SiteMinder WSS Agent requires that LD_LIBRARY_PATH include /usr/lib before any other directory containing older versions of libm.so.
- SHLIB_PATH
- LIBPATH

To set the SiteMinder WSS Agent environment variables after installation, source the following script after you install and configure the SiteMinder WSS Agent:

1. Open a command window.
2. Navigate to *WSS_Home*/webagent/.

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.
3. Enter the following command:

   . ./ca_wa_env.sh

Note: You do not have to run this script for Sun Java System web servers because this file as been added to the start script.

Run the SiteMinder WSS Agent Configuration Program on UNIX or Linux Systems

You can configure your SiteMinder WSS Agents and register a trusted host immediately after installing the SiteMinder WSS Agent or at a later time; however, the host must be registered to communicate with the Policy Server.

Note: You only register the host once, *not* each time you install and configure a SiteMinder WSS Agent on your system.

These instructions are for GUI and Console Mode registration. The steps for the two modes are the same, with the following exceptions for Console mode:

- You may be instructed to select an option by entering a corresponding number for that option.
- You press Enter after each step to proceed through the process. The prompts should guide you through the process.
- All passwords that you enter are displayed in clear text. To workaround this issue, run the installation in GUI or unattended mode.

To configure Agents and register a trusted host

1. If necessary, start the Configuration Wizard as follows:
   a. Open a console window.
   b. Navigate to *agent_home*/install_config_info, where *agent_home* is the installed location of the SiteMinder WSS Agent.
   c. Enter one of the following commands:

      GUI Mode: ./ca-pep-config.bin

      Console Mode: ./ca-pep-config.bin -i console

   The Configuration Wizard starts.

2. Use gathered system and component information to configure the SiteMinder WSS Agent and register the host.

   Note: If you choose to configure multiple Agents, you can set the Register with same Policy Server option to register them all with the same Policy Server.

When the wizard completes, the host is registered and a host configuration file, SmHost.conf, is created in *agent_home*/config. You can modify this file.

agent_home

Is the installed location of the SiteMinder WSS Agent

(Optional) Run the Unattended or Silent Installation and Configuration Programs for your SiteMinder WSS Agent

The unattended or silent installation option can help you automate the installation and configuration process. This method saves time if you have a large CA SiteMinder® Web Services Security environment that uses many agents with identical settings.

For example, suppose the Agents in your environment use the same web server version, installation directory, Agent Configuration Object and Policy Servers. Use the installation wizard or console-based installation program for your first installation. Afterwards, you could create your own script to run the installation program with the .properties file the wizard or console-based installation program created.

Follow these steps:

1. Run the following wizards on your first web server (in the order shown):
   a. The CA SiteMinder® Web Services Security Installation wizard.
   b. The CA SiteMinder® Web Services Security Configuration wizard.
2. Locate the following file on your first web server:

   WSS_Home`/install_config_info/ca-wss-installer.properties`

   Note: If the path contains spaces, surround it with quotes.

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.

3. Perform each of the following steps on the subsequent web servers:

   Note: To automate this process, create your own customized script to execute these files on your systems. Use any scripting language that you want.

   a. Create a temporary directory on the subsequent web server.

   b. Copy the following files from the web server where you ran the wizards (from Steps 1 and 2) to the temporary directory on your subsequent web server:

      - The SiteMinder WSS Agent Installation executable file.
      - The ca-pepconfig-installer.properties file.

   c. Open a Command Prompt window with root privileges in the temporary directory.

   d. Run the following command:

      `ca-sm-wss-12.51-cr-unix_version.bin -f properties_file -i silent`

      cr

      Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

      The SiteMinder WSS Agent is installed and configured on the web server silently.

   e. (Optional) Delete the temporary directory from your web server.

4. Repeat Step 3 for each additional web server in your CA SiteMinder® environment that uses the configuration that the settings in your ca-wss-installer.properties file specify.

Apply CA SiteMinder® Changes to Oracle iPlanet Configuration Files with Oracle iPlanet Administration Server Console for SunOne 6.1 Servers

The Agent Configuration Wizard modifies the default obj.conf, and mime.types files that the Oracle iPlanet web server uses.

If you are using version 6.1 of a SunOne web server, and you plan to use the Oracle iPlanet Administration console, apply the changes to these files *before* using the console. If you do not apply the changes using the console first, the changes that are made for your CA SiteMinder® configuration could be corrupted. If you lose your configuration, run the configuration program again.

Note: The agent adds settings to the obj.conf file of the Oracle iPlanet web server when the Agent is configured to support an advanced authentication scheme. CA SiteMinder® does *not* remove these settings later. Edit the obj.conf file manually to remove any obsolete settings.

Follow these steps:

1. Log in to the Oracle iPlanet Administration Server console.
2. From the Servers tab, select the web server with the CA SiteMinder® agent installed and click Manage.
3. In the right corner of the dialog, click Apply.

   A warning message about loading the modified configuration files appears.
4. Click Load Configuration Files.
5. Exit the console.
6. Restart the web server.
7. Optimize the Agent for Oracle iPlanet by tuning the shared memory segments.

   The CA SiteMinder® changes are applied.

More information:

Reconfigured Web Agent Won't Operate (see page 114)

Manually Configure Non-Default Server Instances, Virtual Servers, or Reverse Proxies for Oracle iPlanet Web Servers

The SiteMinder WSS Agent Configuration wizard only configures the default instance of your Oracle iPlanet web server. To configure a different instance of the Oracle iPlanet web server for CA SiteMinder®, manually edit the obj.conf file that is associated with that server instance. Examples of server instances that need manual configuration include:

- Servers installed in a nondefault directory
- Servers that you want to configure as a reverse proxy. We recommend configuring the reverse proxy using your Oracle iPlanet interface *before* adding the CA SiteMinder® settings to the obj.conf file.

Note: The CA SiteMinder® Agent Configuration wizard *only* modifies the *default* obj.conf file on the Oracle iPlanet (formerly Sun Java System) web server. To protect other instances or reverse proxy deployments with CA SiteMinder®, copy the CA SiteMinder® settings from the default obj.conf file to any respective *instance_name*-obj.conf files. For example, your web server created an obj.conf file when you installed it, but you later added a server instance named my_server.example.com. To protect resources on my_server.example.com with CA SiteMinder®, copy the CA SiteMinder® settings the wizard added from the obj.conf file to the my_server.example.com-obj.conf file.

- Virtual servers on the same computer

Note: SunOne/Sun Java 7.0 web servers do *not* require these manual configuration steps.

Follow these steps:

1. Locate the directory of the server instance you want to configure.
2. Open the obj.conf file with a text editor.
3. Locate the following line:

   ```
   <Object name="default">
   ```

4. Insert a new line below the previous one, and then add the following text:

   ```
   AuthTrans fn="SiteMinderAgent"
   ```

5. Locate the following line:

   ```
   AuthTrans fn="match-browser" browser="*MSIE*" ssl-unclean-shutdown="true"
   ```

6. Insert a new line below the previous one, and then add the following text:

   ```
   NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi" dir="agent_home/pw"
   name="cgi"
   NameTrans fn="pfx2dir" from="/siteminderagent/pw" dir="agent_home/pw"
   NameTrans fn="pfx2dir" from="/siteminderagent/jpw" dir="agent_home/jpw"
   NameTrans fn="pfx2dir" from="/siteminderagent/redirectjsp"
   dir="agent_home/affwebservices/redirectjsp"
   ```

```
NameTrans fn="pfx2dir" from="/siteminderagent/certoptional"
dir="agent_home/samples"
NameTrans fn="pfx2dir" from="/siteminderagent" dir="agent_home/samples"
NameTrans fn="pfx2dir" from="/siteminderagent/pwservlet" dir=agent_home/jpw"
```

agent_home

Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

7. Locate the following line:

   ```
   NameTrans fn="ntrans-j2ee" name="j2ee"
   ```

8. Insert a new line below the previous one, and then add the following text:

   ```
   PathCheck fn="SmRequireAuth"
   ```

9. Remove the following line:

   ```
   NameTrans fn="pfx2dir" from="/mc-icons" dir="C:/Program
   Files/Sun/WebServer7.0/lib/icons" name="es-internal"
   ```

10. Locate the following line:

    ```
    ObjectType fn="force-type" type="text/plain"
    ```

11. Insert a new line below the previous one, and then add the following text:

    ```
    Service method="(GET|POST)" fn="SmAdvancedAuth"
    ```

12. Locate the following line:

    ```
    Error fn="error-j2ee
    ```

13. Insert a new line above the previous one, and then add the following text:

    ```
    Error fn="SmSoapFault" code="500" reason="SmSoapFault"
    ```

14. Save the obj.conf file.
15. Open the magnus.conf file with a text editor.
16. Add the following line:

    ```
    Init fn="load-modules" shlib="agent_home/bin/SunOneWebAgent.dll"
    funcs="SmInitAgent,SmInitChild,SiteMinderAgent,SmRequireAuth,SmAdvancedAuth,S
    mSoapFault
    ```

17. Save the magnus.conf file.

    The Oracle iPlanet web server is manually configured.

Modify the Oracle iPlanet Startup Script to Prevent Crashes when the Server Stops

The Oracle iPlanet server can sometimes crash when shutting down in the following operating environments:

- Solaris 9 SP3
- Solaris 10

Modify the startserv script to prevent the Oracle iPlanet web server from crashing when shuttng down.

Follow these steps:

1. Open the following file with a text editor:

   sunone_instance_directory/bin/startserv

   sunone_instance_directory

   Indicates the directory of the SunOne web server instance.

2. Locate the following line:

   ```
   LIBUMEM_32=/usr/lib/libumem.so
   ```

3. Add a comment character in the beginning of the previous line. See the following example:

   ```
   #LIBUMEM_32=/usr/lib/libumem.so
   ```

4. Locate the following line:

   ```
   LIBUMEM_64=/usr/lib/64/libumem.so
   ```

5. Add a comment character in the beginning of the previous line. See the following example:

   ```
   #LIBUMEM_64=/usr/lib/64/libumem.so
   ```

6. Save the file and close the text editor.

   The Oracle iPlanet startup script is modified.

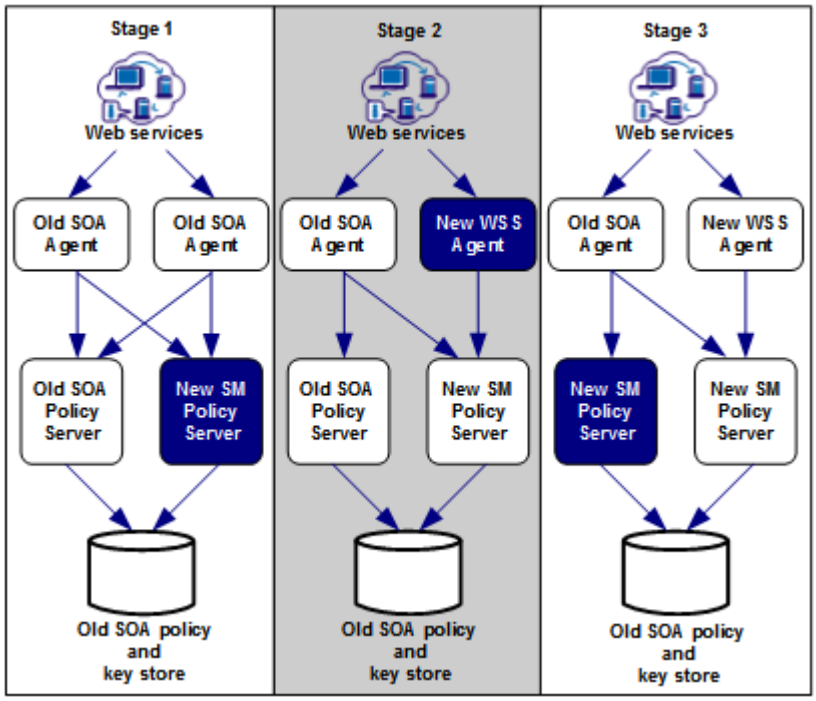
Chapter 5: Upgrade a SOA Agent to a 12.51 WSS Agent

This section contains the following topics:

How to Upgrade a SOA Agent

Upgrading a SOA Agent to a 12.51 WSS Agent involves several separate procedures. To upgrade your agent, Follow these steps::

1. Verify that you are in the proper step of the upgrade process for an agent upgrade. You upgrade agents to 12.51 from r12.1 SP3 at stage two of the CA SiteMinder® Web Services Security upgrade process, as shown in the following illustration:



2. Create backup copies of any customized agent-related files on your web server. Examples of files you could have customized after installing or configuring your agent include the following files:
   - LocalConfig.conf
   - WebAgent.conf
3. If you are upgrading an agent on a UNIX/Linux operating environment, clear the LD_PRELOAD variable (see page 56).
4. Gather information for the following CA SiteMinder® programs.
   - Agent installation wizard.
   - Agent configuration wizard.
5. Run the installation wizard to upgrade your agent on Windows (see page 57) or UNIX (see page 58).
6. If you are upgrading an agent on a UNIX/Linux operating environment, source the agent environment script on the upgraded agent (see page 46)).
7. Run the configuration wizard to configure the upgraded agent on Windows (see page 59) or UNIX (see page 60).
8. If you plan to use the Oracle iPlanet Administration console, apply the changes to your upgraded CA SiteMinder® configuration files (see page 61).
9. Manually configure any nondefault Oracle iPlanet server instances (see page 33).

Verify That the LD_PRELOAD Variable Does Not Conflict with Existing Agent

If you are upgrading or reinstalling a SiteMinder WSS Agent on a Linux system, from the shell, set the LD_PRELOAD variable so that it points to a different location from any existing agent installation directory. For example, if an existing LD_PRELOAD entry is set to:

LD_PRELOAD=*agent_home*/bin/libbtunicode.so

Before you reinstall or upgrade, set the variable to:

export LD_PRELOAD=

This entry sets the variable to a blank value.

Run the Installation Wizard to Upgrade Your Agent on Windows

The installation program for the SiteMinder WSS Agent installs the agent on one computer at a time using the Windows operating environment. This installation program can be run in wizard or console modes. The wizard and console-based installation programs also create a .properties file for subsequent installations and configurations using the unattended or silent method with the same settings.

For example, suppose the Agents in your environment use the same web server version, installation directory, Agent Configuration Object and Policy Servers. Use the installation wizard or console-based installation program for your first installation. Afterwards, you could create your own script to run the installation program with the .properties file the wizard or console-based installation program created.

Follow these steps:

1. Copy the SiteMinder WSS Agent installation executable file to a temporary directory on your web server.
2. Do *one* of the following steps:
   - For wizard-based installations, right-click ca-sm-wss-<SVMVER>-*cr*-win32.exe, and then select Run as Administrator.

     cr

     Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.
   - For console-based installations, open a command line window and run the executable as shown in the following example:

     ```
     ca-sm-wss-<SVMVER>-cr-win32.exe -i console
     ```

   Important! If you are running this wizard on Windows Server 2008, run the executable file with Administrator permissions, even if you are logged into the system as an Administrator. For more information, see the CA SiteMinder® Web Services Security Release Notes.
3. Use the information that you gathered previously to complete the installation.

Note: The software upgrade occurs in the installed location of the existing SOA Agent.

Run the Installation Wizard to Upgrade your Agent on UNIX/Linux

The installation program for the SiteMinder WSS Agent installs the agent on one computer at a time using the UNIX or Linux operating environments. This installation program can be run in wizard or console modes. The wizard and console-based installation program also creates a .properties file for subsequent installations and configurations using the unattended or silent method with the same settings.

For example, suppose the Agents in your environment use the same web server version, installation directory, Agent Configuration Object and Policy Servers. Use the installation wizard or console-based installation program for your first installation. Afterwards, you could create your own script to run the installation program with the .properties file the wizard or console-based installation program created.

Follow these steps:

1. Copy the SiteMinder WSS Agent installation executable file to a temporary directory on your web server.
2. Log in as a root user.
3. Do *one* of the following steps:
   - For wizard-based installations, run ca-sm-wss-<SVMVER>-*cr-unix_version*.bin

     cr

     Specifies the cumulative release number. The base 12.51 release does not include a cumulative release number.

     unix_version

     Specifies the UNIX version: **sol** or **linux**...
   - For console-based installations, open a command-line window and run the executable as shown in the following example:

     ```
     ca-sm-wss-<SVMVER>-cr-unix_version.bin -i console
     ```
4. Use the information from your agent Installation worksheet to complete the installation program.

Note: The software upgrade occurs in the installed location of the existing SOA Agent.

Set Environment Variables for a SiteMinder WSS Agent on UNIX

After installing the SiteMinder WSS Agent on UNIX, you must set required environment variables using the ca_wa_env.sh script. Running the script for SiteMinder WSS Agents on most UNIX platforms ensures that the SiteMinder WSS Agent and web server can work together.

The ca_wa_env.sh script sets the following environment variables:

- NETE_WA_ROOT
- PATH
- NETE_WA_PATH
- LD_LIBRARY_PATH

  Note: The SiteMinder WSS Agent requires that LD_LIBRARY_PATH include /usr/lib before any other directory containing older versions of libm.so.
- SHLIB_PATH
- LIBPATH

To set the SiteMinder WSS Agent environment variables after installation, source the following script after you install and configure the SiteMinder WSS Agent:

1. Open a command window.
2. Navigate to *WSS_Home*/webagent/.

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.
3. Enter the following command:

   . ./ca_wa_env.sh

Note: You do not have to run this script for Sun Java System web servers because this file as been added to the start script.

Run the Configuration Wizard on Your Upgraded SiteMinder WSS Agent on Windows

After gathering the information for your agent configuration, run the agent configuration program. This program creates an agent runtime instance for the web servers running on your computer.

This configuration program is wizard or console based, depending on the option you select. Running the configuration program in the wizard or console mode once creates a properties file. Use the properties file to run unattended configurations on other computers with same operating environment in the future.

Follow these steps:

1. Open the following directory on your web server:

   WSS_Home\install_config_info

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.

   Default: C:\Program Files\CA\Web Services Security

2. Use *one* of the following configuration methods:

   - For a GUI-based configuration, right-click ca-pep-config.exe, and then select Run as Administrator:
   - For a console-based configuration, enter the following command from a Command Prompt window with Administrator privileges open to *WSS_Home*\install_config_info:

     ```
     ca-pep-config.exe -i console
     ```

     Important! If you are running this wizard on Windows Server 2008, run the executable file with administrator permissions. Use these permissions even if you are logged in to the system as an administrator. For more information, see the release notes for your CA SiteMinder® component.

3. Use the information you gathered earlier to complete the wizard.
4. The agent runtime instance is created for your web servers.

Run the Configuration Wizard on Your Upgraded SiteMinder WSS Agent on UNIX/Linux

After gathering the information for your agent configuration, run the agent configuration program. This program creates an agent runtime instance for the web servers running on your computer.

This configuration program is wizard or console based, depending on the option you select. Running the configuration program in the wizard or console mode once creates a properties file. Use the properties file to run unattended configurations on other computers with same operating environment in the future.

Follow these steps:

1. Open a Console Window with root privileges on your web server:
2. Navigate to the following location:

   WSS_Home/install_config_info

   WSS_Home

   Specifies the path to where CA SiteMinder® Web Services Security is installed.

3. Enter one of the following commands:

   GUI Mode: ./ca-pep-config.bin

   Console Mode: ./ca-pep-config.bin -i console

   The Configuration Wizard starts.

4. Use *one* of the following configuration methods:

   - For a GUI-based configuration, run ca-pep-config.bin.
   - For a console-based configuration, open a Command prompt window with root privileges and enter the following command:

     ```
     ca-pep-config.exe -i console
     ```

5. Use the information you gathered earlier to complete the wizard.

   The agent runtime instance is created for your web servers.

Apply Changes to your Upgraded CA SiteMinder® Files with the iPlanet Administration Console

The Agent Configuration Wizard modifies the default obj.conf, and mime.types files that the Oracle iPlanet web server uses.

If you are using version 6.1 of a SunOne web server, and you plan to use the Oracle iPlanet Administration console, apply the changes to these files *before* using the console. If you do not apply the changes using the console first, the changes that are made for your CA SiteMinder® configuration could be corrupted. If you lose your configuration, run the configuration program again.

Note: The agent adds settings to the obj.conf file of the Oracle iPlanet web server when the Agent is configured to support an advanced authentication scheme. CA SiteMinder® does *not* remove these settings later. Edit the obj.conf file manually to remove any obsolete settings.

Follow these steps:

1. Log in to the Oracle iPlanet Administration Server console.
2. From the Servers tab, select the web server with the CA SiteMinder® agent installed and click Manage.
3. In the right corner of the dialog, click Apply.

   A warning message about loading the modified configuration files appears.

4. Click Load Configuration Files.

5. Exit the console.
6. Restart the web server.
7. Optimize the Agent for Oracle iPlanet by tuning the shared memory segments.

   The CA SiteMinder® changes are applied.

Manually Configure Non-Default Server Instances, Virtual Servers, or Reverse Proxies for Oracle iPlanet Web Servers

The SiteMinder WSS Agent Configuration wizard only configures the default instance of your Oracle iPlanet web server. To configure a different instance of the Oracle iPlanet web server for CA SiteMinder®, manually edit the obj.conf file that is associated with that server instance. Examples of server instances that need manual configuration include:

- Servers installed in a nondefault directory
- Servers that you want to configure as a reverse proxy. We recommend configuring the reverse proxy using your Oracle iPlanet interface *before* adding the CA SiteMinder® settings to the obj.conf file.

  Note: The CA SiteMinder® Agent Configuration wizard *only* modifies the *default* obj.conf file on the Oracle iPlanet (formerly Sun Java System) web server. To protect other instances or reverse proxy deployments with CA SiteMinder®, copy the CA SiteMinder® settings from the default obj.conf file to any respective *instance_name*-obj.conf files. For example, your web server created an obj.conf file when you installed it, but you later added a server instance named my_server.example.com. To protect resources on my_server.example.com with CA SiteMinder®, copy the CA SiteMinder® settings the wizard added from the obj.conf file to the my_server.example.com-obj.conf file.
- Virtual servers on the same computer

Note: SunOne/Sun Java 7.0 web servers do *not* require these manual configuration steps.

Follow these steps:

1. Locate the directory of the server instance you want to configure.
2. Open the obj.conf file with a text editor.
3. Locate the following line:

   ```
   <Object name="default">
   ```

4. Insert a new line below the previous one, and then add the following text:

   ```
   AuthTrans fn="SiteMinderAgent"
   ```

5. Locate the following line:

   ```
   AuthTrans fn="match-browser" browser="*MSIE*" ssl-unclean-shutdown="true"
   ```

6. Insert a new line below the previous one, and then add the following text:

   ```
   NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi" dir="agent_home/pw"
   name="cgi"
   NameTrans fn="pfx2dir" from="/siteminderagent/pw" dir="agent_home/pw"
   NameTrans fn="pfx2dir" from="/siteminderagent/jpw" dir="agent_home/jpw"
   NameTrans fn="pfx2dir" from="/siteminderagent/redirectjsp"
   dir="agent_home/affwebservices/redirectjsp"
   NameTrans fn="pfx2dir" from="/siteminderagent/certoptional"
   dir="agent_home/samples"
   NameTrans fn="pfx2dir" from="/siteminderagent" dir="agent_home/samples"
   NameTrans fn="pfx2dir" from="/siteminderagent/pwservlet" dir=agent_home/jpw"
   ```

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

   Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

   Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

   Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

7. Locate the following line:

   ```
   NameTrans fn="ntrans-j2ee" name="j2ee"
   ```

8. Insert a new line below the previous one, and then add the following text:

   ```
   PathCheck fn="SmRequireAuth"
   ```

9. Remove the following line:

   ```
   NameTrans fn="pfx2dir" from="/mc-icons" dir="C:/Program
   Files/Sun/WebServer7.0/lib/icons" name="es-internal"
   ```

10. Locate the following line:

    ```
    ObjectType fn="force-type" type="text/plain"
    ```

11. Insert a new line below the previous one, and then add the following text:

    ```
    Service method="(GET|POST)" fn="SmAdvancedAuth"
    ```

12. Locate the following line:

    ```
    Error fn="error-j2ee
    ```

13. Insert a new line above the previous one, and then add the following text:

    ```
    Error fn="SmSoapFault" code="500" reason="SmSoapFault"
    ```

14. Save the obj.conf file.
15. Open the magnus.conf file with a text editor.
16. Add the following line:

    ```
    Init fn="load-modules" shlib="agent_home/bin/SunOneWebAgent.dll"
    funcs="SmInitAgent,SmInitChild,SiteMinderAgent,SmRequireAuth,SmAdvancedAuth,S
    mSoapFault
    ```

17. Save the magnus.conf file.

    The Oracle iPlanet web server is manually configured.

Chapter 6: Advanced Configuration

This section contains the following topics:

SiteMinder WSS Agent Configuration Parameters

The following table lists configuration parameters for the SiteMinder WSS Agent.

Parameter Name	Value	Description
IgnoreXMLSDK	yes or no	If this parameter is added to the Agent Configuration Object and is set to Yes, the SiteMinder WSS Agent is disabled. This means that the Agent behaves as a Web Agent for all incoming requests. If added to the Agent Configuration Object and set to No (or not added to to the Agent Configuration Object at all), the SiteMinder WSS Agent is enabled. That is, the Agent uses the XML SDK to process incoming HTTP requests under these conditions: ■ HTTP action is POST ■ HTTP MIME type is “text/xml” or, if the XMLSDKMimeTypes parameter is configured, any one of the MIME types specified by that parameter. ■ HTTP content is an XML document

Parameter Name	Value	Description
XMLSDKMimeTypes	String	A comma-delimited list of MIME types that the SiteMinder WSS Agent will accept for processing by CA SiteMinder® Web Services Security. All POSTed requests having one of the listed MIME types are processed. Examples: ■ text/xml ■ application/octet-stream ■ text/xml,multipart/related If you do not add this parameter to the Agent Configuration Object, the SiteMinder WSS Agent defaults to accepting only the text/xml MIME type.
ServerProductName	String	Description of the product name—for example, iPlanet Web Server. Provides a value for the SiteMinder WSS Agent variable property Server Product Name. **Note:** For more information about setting this variable, see the *CA SiteMinder® Web Services Security Policy Configuration Guide*.
ServerVendor	String	Description of the Web Server vendor—for example, Sun. Provides a value for the SiteMinder WSS Agent Variable property Server Vendor. **Note:** For more information about setting this variable, see the *CA SiteMinder® Web Services Security Policy Configuration Guide*.
ServerVersion	String	Description of the product version—for example, 6.0 SP2). Provides a value for the SiteMinder WSS Agent Variable property Server Version. **Note:** For more information about setting this variable, see the *CA SiteMinder® Web Services Security Policy Configuration Guide*.

Parameter Name	Value	Description
MaxXmlSdkRetries	Number	Defines the number of times the SiteMinder WSS Agent tries to contact the XML SDK server when it receives requests. The default is 3. The Agent does not continually retry the server for the same request. If a request comes in and the Agent cannot contact the SDK server, that request is dropped and the Agent tries again when a subsequent request is made. The Agent attempts to connect for each new request until it reaches the number specified in this parameter. If the SiteMinder WSS Agent does not connect to the XML SDK server, the Agent assumes the server is not running and stops trying to process CA SiteMinder® Web Services Security-specific requests. **Note:** For the SiteMinder WSS Agent on Apache Web servers, this value applies to each process.
XMLSDKResourceIdentification	yes or no	Determines if the SiteMinder WSS Agent should identify the web service operation being requested by an incoming XML message as well as the resource identifier (that is, perform fine-grain resource identification). The default is No. **Note:** You must set this option to Yes if you want to use the Administrative UI to configure policies to protect resources with the SiteMinder WSS Agent.
XMLSDKAcceptSMSessionCookie	yes or no	Determines whether or not the SiteMinder WSS Agent accepts an CA SiteMinder session cookie to authenticate a client. The default is no. If set to yes, the SiteMinder WSS Agent uses information in a session cookie sent as an HTTP header in the request as a means of authenticating the client. If set to no, session cookies are ignored and the SiteMinder WSS Agent requests credentials required by the configured authentication scheme.

Parameter Name	Value	Description
SAMLSessionTicketLogoffi	yes or no	Determines whether the SiteMinder WSS Agent should attempt to log off session tickets in SAML assertions. The default is yes.
XMLAgentSoapFaultDetails	yes or no	Determines whether or not the SiteMinder WSS Agent should insert the authentication/authorization rejection reason (if provided by the Policy Server) into the SOAP fault response sent to the Web service consumer. The default is no.

Configure a SiteMinder WSS Agent to Enable Fine-Grain Resource Identification

By default, the SiteMinder WSS Agent identifies incoming requests for web service resources as follows:

[*URL*][*Web Service Name*]

However, the SiteMinder WSS Agent can be configured to provide fine-grain resource identification, additionally identifying the requested web service operation name, so that requests are identified as:

[*URL*][*Web Service Name*][*Web Service Operation*]

This allows you to define fine-grain policies that include the web service operation in authorization decisions, but may adversely affect transaction performance.

Note: For more information on configuring fine-grain authorization policies, see the *CA SiteMinder® Web Services Security Policy Configuration Guide*.

Follow these steps:

1. Ensure that the XMLSDKResourceIdentification Agent configuration parameter is present and set to Yes for the target SiteMinder WSS Agent.
2. Edit the XmlToolkit.properties file located in *agent_home*\java to ensure that the WSDMResourceIdentification entry is present and set to "Yes".

3. Save and close the XmlToolkit.properties file.
4. Restart the target SiteMinder WSS Agent.

Note: You must enable fine-grain resource identification to use the Administrative UI to generate policies for web service resources protected by the SiteMinder WSS Agent from their associated WSDL files.

Configure the Username and Password Digest Token Age Restriction

By default, the WS-Security authentication scheme imposes a 60-minute restriction on the age of Username and Password Digest Tokens to protect against replay attacks.

To configure a different value for the token age restriction for a SiteMinder WSS Agent for Web Servers, add the WS_UT_CREATION_EXPIRATION_MINUTES parameter to the XmlToolkit.properties file for that agent.

Follow these steps:

1. Navigate to a*gent_home*\java.
2. Open XmlToolkit.properties in a text editor.
3. Add the following line:

   `WS_UT_CREATION_EXPIRATION_MINUTES=`*`token_age_limit`*

   token_age_limit

   Specifies the token age limit restriction in minutes.
4. Save and close the XmlToolkit.properties file.
5. Restart the SiteMinder WSS Agent.

Configure the SiteMinder WSS Agent to Process Large XML Messages

By default, the SiteMinder WSS Agent can process XML messages up to 2000 KB in size. However, if you need to process larger files, you can increase this size limit by editing the conapi.conf file on each system hosting a SiteMinder WSS Agent.

The conapi.conf is located in:

- *agent_home*\config (Windows)
- *agent_home*/config (UNIX)

agent_home

Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

To increase the maximum message size, uncomment the nete.conapi.service.xmlsdk.maxpacketsize entry and change its value to the maximum message size (in KB) you want the SiteMinder WSS Agent to be able to process. For example:

```
nete.conapi.service.xmlsdk.maxpacketsize=3000
```

Note: Web servers also have incoming file size limits. When planning your Web service implementations, you should ensure that you are aware of these and bear them in mind.

Oracle iPlanet Web Server Settings

Use any of the following settings to manage your CA SiteMinder® Agent Oracle iPlanet servers:

- Restrict directory browsing (see page 71).
- Handle multiple AuthTrans functions (see page 71).
- Record the transaction ID in the Oracle iPlanet web server log (see page 72).

Restrict Directory Browsing on an Oracle iPlanet Web Server

To help ensure that users who try to browse the directories of a Oracle iPlanet web server are challenged by CA SiteMinder®, you can set the following parameter:

DisableDirectoryList

Specifies whether the Web Agent allows a user to view or browse the contents of a directory without challenging them first. This occurs when *all* of the following conditions are true:

- The realm is set to protect a root resource (/)
- The default web page in the directory (such as index.html) is renamed or deleted.

Default: No

To restrict directory browsing on a Oracle iPlanet server

1. Add the DisableDirectoryList parameter to your Agent Configuration object or your local configuration file.
2. Set the value of the DisableDirectoryList parameter to yes.

   Directory browsing is restricted. CA SiteMinder® challenges users who try to browse directories.

Handle Multiple AuthTrans Functions for Oracle iPlanet Web Servers

AuthTrans functions are directives that initialize the Oracle iPlanet web server. The Oracle iPlanet web server executes AuthTrans functions in the order that they are listed in the obj.conf file. The Oracle iPlanet server reads through the AuthTrans functions until it finds a function that returns a REQ_PROCEED command. Once a REQ_PROCEED command executes, no other AuthTrans functions are executed.

By default, CA SiteMinder® is the first AuthTrans function and it returns a REQ_PROCEED. To allow other AuthTrans functions to execute, you need to add the EnableOtherAuthTrans parameter and set the value to yes.

The default value for this parameter is no. To enable multiple AuthTrans functions set the EnableOtherAuthTrans parameter to yes.

By adding this parameter, you permit the CA SiteMinder® Web Agent to exist with other functions.

Be sure the CA SiteMinder® Agent function is the first entry in the obj.conf file for the AuthTrans directive. The entry should read:

```
AuthTrans fn="SiteMinderAgent"
```

Record the Transaction ID in Oracle iPlanet Web Server Logs

Valid on Solaris

The Web Agent generates a unique transaction ID for each successful user authorization request. The Agent adds the ID to the HTTP header. The ID is also recorded in the following logs:

- Audit log
- Web server log (if the server is configured to log query strings)
- Policy Server log

You can track user activities for a given application using the transaction ID.

Note: For more information, see the Policy Server documentation.

The transaction ID appears in the log as a mock query parameter in the log that is appended to the end of an existing query string. The following example shows transaction ID (in bold) appended to a query string (which ends with STATE=MA):

```
172.24.12.1, user1, 2/11/00, 15:30:10, W3SVC, MYSERVER, 192.168.100.100, 26844,
47, 101, 400, 123, GET, /realm/index.html,
STATE=MA&SMTRANSACTIONID=0c01a8c0-01f0-38a47152-01ad-02714ae1
```

If no query parameters are in the URL, the Agent adds the transaction ID at the end of the web server log entry. For example:

```
172.24.12.1, user1, 2/11/00, 15:30:10, W3SVC, MYSERVER, 192.168.100.100, 26844,
47, 101, 400, 123, GET, /realma/index.html,
SMTRANSACTIONID=0c01a8c0-01f0-38a47152-01ad-02714ae1.
```

Note: Web Agents log user names and access information in native web server log files when users access resources.

You can record the CA SiteMinder® transaction ID in the Oracle iPlanet web server logs.

Follow these steps:

1. Open the magnus.conf file.
2. Add the following header variable to the existing list of HTTP server variables that you want to log when the web server initializes:

   ```
   %Req->headers.SM_TRANSACTIONID%"
   ```

   Note: Enter the header variable in uppercase unless the value of the LowerCaseHTTP parameter is set to yes in your Agent Configuration Object or local configuration file.

   The following example shows the SMTRANSACTIONID header variable in bold at the end of an existing entry. However, you can place it anywhere in the list of variables.

   ```
   Init fn="flex-init" access="D:/iPlanet/server4/https-orion/logs/access"
   format.access="%Ses->client.ip% - %Req->vars.auth-user% [%SYSDATE%] \"
   %Req->srvhdrs.clf-status% %Req-srvhdrs.content-length% %Req->headers.-
   SM_TRANSACTIONID%"
   ```

3. Restart the Oracle iPlanet Server to apply the change.

   The transaction ID appears in the Oracle iPlanet web server logs. The following example shows a web server log entry with the transaction ID in bold:

   ```
   11.22.33.44 - user1 [21/Nov/2003:16:12:24 -0500] "GET /Anon/index.html HTTP/1.0"
   200 748 3890b4b9-58f8-4a74df53-07f6-0002df88
   ```

Chapter 7: Dynamic Policy Server Clusters

Earlier versions of CA SiteMinder® agents did *not* automatically discover when Policy Servers were added or removed from a cluster. The agents recognized the changes only after their respective web servers were restarted.

CA SiteMinder® 12.51 supports dynamic Policy Server clusters. Agents automatically discover Policy Servers that are added or removed from an existing cluster when dynamic Policy Server Clusters are enabled.

For example, suppose that your agent connects to a cluster of the following Policy Servers:

- 192.168.2.100
- 192.168.2.101
- 192.168.2.103
- 192.168.2.104

Suppose that you later decide to remove the server 192.168.2.103 to upgrade its operating system. In this situation, enabling dynamic Policy Server clusters lets your agents recognize the change in the membership of the cluster without restarting.

Restart your web server if you do any of the following tasks:

- Change the configuration of an existing Policy Server (using the configuration wizard).
- Create a Policy Server cluster.
- Delete a Policy Server cluster.
- Change the values for any of the following Policy Server settings:
  - EnableFailOver
  - MaxSocketsPerPort
  - MinSocketsPerPort
  - NewSocketStep
  - RequestTimeout

Connect a SiteMinder WSS Agent to a Dynamic Policy Server Cluster

You can connect a SiteMinder WSS Agent to one or more dynamic Policy Server clusters by modifying the SmHost.conf file on your web server.

Follow these steps:

1. Open the following file with a text editor:

   agent_home\config\SmHost.conf

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

   Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

   Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

   Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

2. Do *one* of the following tasks:

   - If this agent has *never* been connected to dynamic cluster of Policy Servers before, create a line (anywhere in the file) with the following text:

     enableDynamicHCO="YES"

   - If this agent has previously been connected to a dynamic cluster of Policy Servers, change the value of the existing enableDynamicHCO parameter from "NO" to "YES".

3. Save the SmHost.conf file, and then close the text editor.

4. Restart your web server.

   The SiteMinder WSS Agent is connected to dynamic Policy Server clusters.

Chapter 8: Starting and Stopping SiteMinder WSS Agents

This section contains the following topics:

Enable a SiteMinder WSS Agent

Configure your agent parameters and then enable the agent to protect the resources on the web server.

Note: *No* resources are protected until you also define policies in the CA SiteMinder® Policy Server.

Follow these steps:

1. Open one of the following files (as appropriate for your operating system) with a text editor:
   - *agent_home*\bin\WebAgent.conf (Windows)
   - *agent_home/*bin/WebAgent.conf (UNIX)

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.
2. Change the value of the EnableWebAgent parameter to yes.
3. Save and close the WebAgent.conf file.
4. Restart the web server (the web server itself, not the computer on which it runs).

   The SiteMinder WSS Agent is enabled.

Disable a SiteMinder WSS Agent

To stop the SiteMinder WSS Agent from protecting the resources on your web server and stop communicating with the Policy Server, disable the SiteMinder WSS Agent.

Follow these steps:

1. Open one of the following files (as appropriate for your operating system) with a text editor:
   - *agent_home*\bin\WebAgent.conf (Windows)
   - *agent_home/*bin/WebAgent.conf (UNIX)

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.
2. Change the value of the EnableWebAgent parameter to no.
3. Save and close the WebAgent.conf file.
4. Restart the web server (the web server itself, not the computer on which it runs).

   The SiteMinder WSS Agent is disabled.

Starting or Stopping Most Apache-based Agents with the apachectl Command

Starting or stopping most Apache-based agents with the apachectl command on UNIX or Linux operating environments requires setting the environment variables for the product first.

Note: The Apache-based agents do *not* support the apachectl -restart option. This procedure does *not* apply to Apche-based IBM HTTP servers. Use this procedure instead.

Follow these steps:

1. For UNIX/Linux operating environments, set the environment variables by running the following script:

   ```
   ./ca_wa_env.sh
   ```
2. Use *one* of the following commands:

   ```
   apachectl -stop
   ```

   ```
   apachectl -start
   ```

Start and Stop SiteMinder WSS Agent Processing

The CA SiteMinder® Web Services Security XML SDK server is a process that must be running so the SiteMinder WSS Agent can process requests. The XML SDK Server is started automatically at system startup. This section describes how to start and stop it manually.

Note: Do not confuse the XML SDK server with the CA SiteMinder® Web Services Security SDK, which is an API that communicates with the XML SDK server.

Start the CA SiteMinder® Web Services Security XML SDK Server

The CA SiteMinder® Web Services Security XML SDK server process must be running for the SiteMinder WSS Agent to process requests.

To start the CA SiteMinder® Web Services Security XML SDK server on Windows

1. Open the Services dialog.
2. Right-click the TxMinder XML SDK Service entry and then click Start in the menu that opens.

To start the CA SiteMinder® Web Services Security XML SDK server on UNIX

1. Open a command window.
2. Navigate to *agent_home.*

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

   Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

   Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

   Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32
3. Enter the following command:

   ```
   . ./ca_wa_env.sh
   ```
4. Navigate to *agent_home*/bin
5. Enter the following command:

   ```
   tmxmlsdkserver -start
   ```

Stop the CA SiteMinder® Web Services Security XML SDK Server

The CA SiteMinder® Web Services Security XML SDK server process must be running for the SiteMinder WSS Agent to process requests.

To stop the CA SiteMinder® Web Services Security XML SDK server on Windows

1. Open the Services dialog.
2. Right-click TxMinder XML SDK Service entry and then click Stop in the menu that opens.

To stop the CA SiteMinder® Web Services Security XML SDK server on UNIX

1. Open a command window.
2. Navigate to *agent_home.*

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

   Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

   Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

   Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32
3. Enter the following command:

   `. ./ca_wa_env.sh`
4. Navigate to *agent_home*/bin
5. Enter the following command:
   `tmxmlsdkserver -stop`

Chapter 9: Operating System Tuning for Agents

This section contains the following topics:

Tune the Shared Memory Segments

If you install an Oracle iPlanet agent on Solaris systems, tune the shared memory settings of the operating environment for the agent to function correctly. By increasing the shared memory segments or your operating environment, you improve the performance of the Agent. The variables that control shared memory segments are defined in the specification file of your operating environment.

For AIX operating environments, run the following command before starting an Oracle iPlanet web server:

```
export EXTSHM=ON
```

Note: Sometimes Linux operating environments require tuning the shared memory segments. For more information about the shared memory segments and how to tune them, see the documentation for your particular operating environment.

Follow these steps:

1. Follow the appropriate procedure for your operating environment:
   - Solaris**:** Open the /etc/system file, using the editor of your choice.
2. Modify the shared memory variables using *one* of the following methods:
   - Solaris: Add the variables shown in the following list and configure them using the recommended settings shown in the examples. Use the following syntax:

     ```
     set shmsys:shminfo_shmmax=33554432
     ```

     shmsys:shminfo_shmmax

     Specifies the maximum shared memory segment size. Controls the maximum size of the Agent resource and session cache.

     Note: To estimate the amount of memory segments that are required, allocate 4 KBs per entry in each cache, or view cache usage statistics in the OneView Monitor. See the *Web Agent Configuration* Guide for more information about using the OneView Monitor.

     Example: 33554432 (32 MB) for busy sites that require large caches.

     shmsys:shminfo_shmmin

     (Not required for Solaris) Minimum shared memory segment size. Controls the minimum size of the Agent resource and session cache.

shmsys:shminfo_shmmni

Specifies the maximum number of shared memory segments that can exist simultaneously, systemwide.

Example: (except Solaris 9) N/A

Example: (Solaris 9) 200

shmsys:shminfo_shmseg

(Not required for Solaris 9) Specifies the maximum number of shared memory segments per process.

Example: 24

semsys:seminfo_semmni

Specifies the number of semaphore identifiers. Use 11 for every instance of the Agent that you run on the system.

Example: (except Solaris 9) 100

Example: (Solaris 9) 200

semsys:seminfo_semmns

Specifies the number of semaphores in the system. Use 10 for every instance of the Agent that you run on the system.

Example: (Solaris 9) 100

Example: (Solaris 9) 400

semsys:seminfo_semmnu

Specifies the number of processes using the undo facility. For optimal performance, set the semmnu value greater than the number of Oracle iPlanet web server processes running on the system at any one time. Exceed the maxprocs setting by 200 or more.

Example: (Solaris 9) 200

3. Save your changes then exit the file or the utility.
4. Reboot the system.
5. Verify your changes by entering the command:
   ```
   $ sysdef -i
   ```

How to Tune the Solaris 10 Resource Controls

Tune the resource controls at the project level to improve the performance of the agent.

Note: See your Solaris documentation for more information.

Tuning the resource controls on Solaris 10 uses the following process:

1. Determine the project that is associated with the user account under which the Web Agent runs.
2. Increase the settings for any of the following resource controls of that project:

   project.max-shm-ids

   Specifies the maximum shared memory IDs for a project.

   project.max-sem-ids

   Specifies the maximum number of semaphore IDs for a project.

   project.max-msg-ids

   Specifies the maximum number of message queue IDs for a project.

   project.max-shm-memory

   Specifies the total amount of shared memory allowed for a project.

   process.max-sem-nsems

   Specifies the maximum number of semaphores allowed per semaphore set.

   process.max-sem-ops

   Specifies the maximum number of semaphore operations allowed per semop.

   process.max-msg-messages

   Specifies the maximum number of messages on a message queue.

   process.max-msg-qbytes

   Specifies the maximum number of bytes of messages on a message queue.

Chapter 10: Uninstall a SiteMinder WSS Agent

This section contains the following topics:

Set JRE in PATH Variable Before Uninstalling the CA SiteMinder® Agent

On Windows and UNIX systems, when you are uninstalling a CA SiteMinder® Agent, make sure the JRE is in the PATH variable or 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."

Follow these steps:

On Windows

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.

   For example, C:\j2sdk*version_number*\jre\bin

On UNIX

Run the following commands:

1. PATH=$PATH:*JRE*/bin

   JRE

   Specifies the location of your JRE.

   For example, /usr/bin/j2sdk*version_number*/jre
2. export PATH

Uninstall a SiteMinder WSS Agent

To uninstall a SiteMinder WSS Agent, run the CA SiteMinder® Web Services Security uninstall wizard.

Follow these steps:

1. Navigate to the WSS_*HOME*\install_config_info (Windows) or WSS_*HOME*/install_config_info (UNIX) directory and run the CA SiteMinder® Web Services Security uninstall wizard to remove core CA SiteMinder® Web Services Security components:

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

   WSS_HOME

   Specifies the CA SiteMinder® Web Services Security installation location.

   Important! If you are running this wizard on Windows Server 2008, run the executable file with Administrator permissions, even if you are logged into the system as an Administrator.

   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 CA SiteMinder® Web Services Security components.

4. Restart the server.

Chapter 11: SiteMinder WSS Agent Logging

This section contains the following topics:

Logs of Start-up Events

To assist in debugging, startup events are recorded in a log. Each message may provide clues about the problem. These logs are stored in the following locations:

- On Windows systems, these events are recorded in the Windows Application Event log.
- On UNIX systems, these events are sent to STDERR. Apache servers map STDERR to the Apache error_log file, so these events are also recorded in that log.

Error Logs and Trace Logs

You can use the Web Agent logging function to monitor the performance of the Web Agent and its communication with the Policy Server. The logging feature provides accurate and comprehensive information about the operation of CA SiteMinder® processes to analyze performance and troubleshoot issues.

A log is a record of events that occur during program execution. A log consists of a series of log messages, each one describing some event that occurred during program execution. Log messages are written to log files.

Note: IIS Agents create log files only after the first user request is submitted. Apache 2.0 Web Agents create log files when the Apache server starts.

The Web Agent uses the following log files:

Error log

Contains program and operational-level errors. One example is when the Web Agent cannot communicate with Policy Server. The level of detail output in this log cannot be customized. Error logs contain the following types of messages:

Error messages

Contain program-level errors, which indicate incorrect or abnormal program behavior, or an inability to function as expected due to some external problem, such as a network failure. There are also operational-level errors. This type of error is a failure that prevents the operation from succeeding, such as opening a file or authenticating a user.

Informational messages

Contain messages for the user or administrator that some event has occurred; that is, that a server has started or stopped, or that some action has been taken.

Warning messages

Contain warnings for the user or administrator of some condition or event that is unusual or indicative of a potential problem. This does not necessarily mean there is anything wrong.

Trace log

Contains detailed warning and informational messages, which you can configure. Examples include trace messages and flow state messages. This file also includes data such as header details and cookie variables. Trace logs contain the following messages:

Trace messages

Provide detailed information about program operation for tracing and/or debugging purposes. Trace messages are ordinarily turned off during normal operation. In contrast to informational, warning, and error messages, trace messages are embedded in the source code and can not easily be localized. Moreover, trace messages may include significant data in addition to the message itself; for example, the name of the current user or realm.

You specify the location of both the error and trace log files when you configure the Web Agent. Use the error and trace logs to help solve any issues that may prevent the Web Agent from operating properly.

Note: For Agents on Windows platforms, set the EnableWebAgent parameter to yes to ensure that the Web Agent log gets created. If you leave EnableWebAgent set to no (the default) and set the logging parameters, the Agent log gets created only for Agents on UNIX platforms.

Parameter Values Shown in Log Files

Web Agents list configuration parameters and their values in the Web Agent error log file, but there are differences between the ways that Traditional and Framework agents do this.

Framework agents record the configuration parameters and their values in the log file exactly as you entered them in the Agent Configuration Object or the local configuration file. All of the parameters, including those which may contain an incorrect value, are recorded in the log file.

Traditional agents process the parameter values before recording them. If the parameter has a proper value, the parameter and its value are recorded in the log file. Parameters with incorrect values are *not* recorded in the log file.

Set Up and Enable Error Logging

Error logs require the following settings:

- Logging is enabled.
- A location for the log file is specified.

The parameters that enable error logging and determine options such as appending log data are defined in a local configuration file or an Agent Configuration Object at the Policy Server.

Agents that are installed on an IIS or Apache web servers do not support dynamic configuration of log parameters that are set locally in a local configuration file. The changes take effect when the Agent is restarts. However, these log settings can be stored and updated dynamically in an agent configuration object at the Policy Server.

Note: IIS Agents create log files only after the first user request is submitted. Apache 2.0 Web Agents create log files when the Apache server starts.

Follow these steps:

1. If you do not have a log file already, create a log file and any related directories.
2. Set the value of the LogFile parameter to yes.

   Note: Setting the value of this parameter to yes in a local configuration file of a web server overrides any of the logging settings that are defined on the Policy Server. For example, suppose that the value of this parameter is set to yes in a LocalConfig.conf file. The agent creates log files even though the value of the AllowLocalConfig parameter in the corresponding agent configuration object is set to no. You can also set the related logging parameters in the LocalConfig.conf file also to override any other settings in the agent configuration object.
3. Specify the full path to the error file, including the file name, in any of the following parameters:

   LogFileName

   Specifies the full path (including the file name) of the log file.

   Default: No

   Example: (Windows) *agent_home*\log\WebAgent.log

   Example: (UNIX/LInux)
   /export/iPlanet/servers/https-jsmith/logs/WebAgent.log

   LogFileName32

   Specifies the full path of a log file for a SiteMinder WSS Agent for IIS (on 64-bit Windows operating environments protecting 32-bit applications). The 32-bit applications run in Wow64 mode on the 64-bit Windows operating environment. If logging is enabled but this parameter is not set, the SiteMinder WSS Agent for IIS appends _32 to the log file name.

   Default: No

   Limits: For Windows 64-bit operating environments only. Specify the file name at the end of the path.

   Example: (Windows 64-bit operating environments using Wow64 mode) *agent_home*\log\WebAgent32.log.
4. (Optional) Set the following parameters (in the Agent Configuration Object on the Policy Server or in the local configuration file):

   LogAppend

   Adds new log information to the end of an existing log file. When this parameter is set to no, the entire log file is rewritten each time logging is invoked.

   Default: No

LogFileSize

Specifies the size limit of the log file in megabytes. When the current log file reaches this limit, a new log file is created. The new log file uses one of the following naming conventions:

- For framework agents, the new log file has a sequence number that is appended to the original name. For example, a log file named myfile.log is renamed to myfile.log.1 when the size limit is reached.
- For traditional agents, the new log files are named by appending the date and timestamp to the original name. For example, a log file named myfile.log, is renamed to myfile.log.09-18-2003-16-07-07 when the size limit is reached.

Archive or remove the old files manually.

Default: 0 (no rollover)

Example: 80

LogLocalTime

Specifies whether the logs use Greenwich Mean Time (GMT) or local time. To use GMT, change this setting to no. If this parameter does not exist, the default setting is used.

Default: Yes

If you use a local configuration file, your settings resemble the following example:

```
LogFile="yes"
LogFileName="/export/iPlanet/servers/https-myserver/logs/errors.log"
LogAppend="no"
LogFileSize="80"
LogLocalTime="yes"
```

Error logging is enabled.

Enable Transport Layer Interface (TLI) Logging

When you want to examine the connections between the agent and the Policy Server, enable transport layer interface logging.

To enable TLI logging

1. Add the following environment variable to your web server.

   `SM_TLI_LOG_FILE`

2. Specify a directory and log file name for the value of the variable, as shown in the following example:

   directory_name/log_file_name.log

3. Verify that your agent is enabled.

4. Restart your web server.

   TLI logging is enabled.

Limit the Number of Log Files Saved

You can limit the number of log files that an agent keeps. For example, if you want to save disk space on the system that stores your agent logs, you can limit the number of log files using the following parameter:

LogFilesToKeep

Specifies the number of agent log files that are kept. New log files are created in the following situations:

- When the agent starts.
- When the size limit of the log file (specified by the value of the LogFileSize parameter) is reached.

Changing the value of this parameter does *not* automatically delete any existing logs files which exceed the number that you want to keep. For example, If your system has 500 log files stored, and you decide to keep only 50 of those files, the agent does *not* delete the other 450 files.

Setting the value of this parameter to zero retains all the log files.

Default: 0

Follow these steps:

1. Archive or delete any existing log files from your system.
2. Set the value of the LogAppend parameter to no.
3. Change the value of the LogFilesToKeep parameter to the number of log files that you want to keep.

How to Set Up Trace Logging

To set up trace logging, use the following process:

1. Set up and Enable Trace logging.
2. Determine what you want to record in the trace log by reviewing the following lists:
   - Trace Log Components and Subcomponents
   - Trace Message Data Fields
   - Data Field Filters
3. Duplicate the default Trace Configuration File.
4. Modify the duplicate file to include the items you want to record.
5. Restart the agent.

Configure Trace Logging

Before you can use trace logging, you must configure it by specifying a name, location, and parameters for the trace log file. These settings control the size and format of the file itself. After trace logging is configured, you determine the content of the trace log file separately. This lets you change the types of information contained in your trace log at any time, without changing the parameters of the trace log file itself.

Follow these steps:

1. Locate the WebAgentTrace.conf file on your web server. Duplicate the file.
2. Open your Agent Configuration Object or local configuration file.
3. Set the TraceFile parameter to yes.

   Note: Setting the value of this parameter to yes in a local configuration file of a web server overrides any of the logging settings that are defined on the Policy Server. For example, suppose that the value of this parameter is set to yes in a LocalConfig.conf file. The agent creates log files even though the value of the AllowLocalConfig parameter in the corresponding agent configuration object is set to no. You can also set the related logging parameters in the LocalConfig.conf file also to override any other settings in the agent configuration object.
4. Specify the full path to the trace log files in following parameters:

   TraceFileName

   Specifies the full path to the trace log file.

   Default: No default

   Limits: Specify the file name in this parameter.
   Example: web_agent_home\log\trace.log

5. Specify the full path to the duplicate copies of WebAgentTrace.conf file (you created in Step 1) in the following parameters:

   TraceConfigFile

   Specifies the location of the WebAgentTrace.conf configuration file that determines which components and events to monitor.

   Default: No default

   Example: web_agent_home\config\WebAgentTrace.conf

   Note: This file is not used until the web server is restarted.

6. Define the format of the information in your trace log file by setting the following parameters in your Agent Configuration Object or local configuration file:

   TraceAppend

   Adds new logging information to the end of an existing log file instead of rewriting the entire file each time logging is invoked.

   Default: No

   TraceFormat

   Specifies how the trace file displays the messages. Choose one of the following options:

   - default—uses square brackets [] to enclose the fields.
   - fixed—uses fields with a fixed width.
   - delim—uses a character of your choice to delimit the fields.
   - xml—uses XML-like tags. A DTD or style sheet is not provided with the Web Agent.

   Default: default (square brackets)

TraceDelimiter

Specifies a custom character that separates the fields in the trace file.

Default: No default

Example: |

TraceFileSize

Specifies (in megabytes) the maximum size of a trace file. The Web Agent creates a new file when this limit is reached.

Default: 0 (a new log file is not created)

Example: 20 (MB)

LogLocalTime

Specifies whether the logs use Greenwich Mean Time (GMT) or local time. To use GMT, change this setting to no. If this parameter does not exist, the default setting is used.

Default: Yes

7. Edit the WebAgentTrace.conf file to have agent monitor the activities you want.

   Framework Agents do not support dynamic configuration of log parameters set locally in the Agent configuration file. Consequently, when you modify a parameter, the change does not take effect until you restart the web server. However, these log settings can be stored and updated dynamically if you configure them in an Agent configuration object on the Policy Server.

8. Restart the web server so the agent uses the new trace configuration file.

Trace Log Components and Subcomponents

The CA SiteMinder® Agent can monitor specific CA SiteMinder® components. When you monitor a component, all of the events for that component are recorded in the trace log. Each component has one or more subcomponents that the agent can also monitor. If you do not want the agent to record all of the events for a component, you can specify only those subcomponents you want to monitor instead.

For example, if you want to record only the single sign-on messages for an agent on a web server, you would specify the WebAgent component and the SSO subcomponent.

The following components and subcomponents are available:

AgentFramework

Records all Agent framework messages. (Applies only to framework agents.) The following subcomponents are available:

- Administration
- Filter
- HighLevelAgent
- LowLevelAgent
- LowLevelAgentWP

AffiliateAgent

Records web Agent messages related to the 4.x Affiliate Agent, which is part of Federation Security Services, a separately-purchased product. (Applies only to framework agents.) The following subcomponent is available:

- RequestProcessing

SAMLAgent

Web Agent messages related to the SAML Affiliate Agent. (Applies only to framework agents.) The following subcomponent is available:

- RequestProcessing

WebAgent

Records all Web Agent log messages. Applies to all Agents *except* IIS 6.0 or Apache 2.0 Agents. The following subcomponents are available:

- AgentCore
- Cache
- authentication
- Responses
- Management
- SSO
- Filter

Agent_Functions

Records all Agent API messages. The following subcomponents are available:

- Init
- UnInit
- IsProtected
- Login

- ChangePassword
- Validate
- Logout
- Authorize
- Audit
- FreeAttributes
- UpdateAttributes
- GetSessionVariables
- SetSessionVariables
- DeleteSessionVariables
- Tunnel
- GetConfig
- DoManagement

Agent_Con_Manager

Records messages related to internal processing of the Agent API. The following subcomponents are available:

- RequestHandler
- Cluster
- Server
- WaitQueue
- Management
- Statistics

Trace Message Data Fields

You can define what each trace message for a specific component contains by specifying which data fields to include in the message.

Data fields use the following syntax:

`data:`*`data_field1`*`,`*`data_field2`*`,`*`data_field3`*

Some data fields are shown in the following example:

```
data:message,date,time,user,agentname,IPAddr
```

There may not be data for fields in each message, so blank fields my occur. For example, if you select RealmOID as a data field, some trace messages will display the realm's OID while others will not.

The following data fields are available:

Message

Includes the actual trace message

SrcFile

Includes the source file and line number of the trace message

Pid

Includes the process ID

Tid

Includes the thread ID

Date

Includes the date

Time

Includes the time

PreciseTime

Includes the time, including milliseconds

Function

Includes the function in the code containing the trace message

User

Includes the name of the user

Domain

Includes the CA SiteMinder® domain

Realm

Includes the CA SiteMinder® realm

AgentName

Includes the Agent name being used

TransactionID

Includes the transaction ID

DomainOID

Includes the CA SiteMinder® domain OID

IPAddr

Includes the client IP address

RequestIPAddr

Includes the trace file displays the IP of the server where Agent is present

IPPort

Includes the client IP port

CertSerial

Includes the certificate serial number

SubjectDN

Includes the subject DN of the certificate

IssuerDN

Includes the Issuer DN of the certificate

SessionSpec

Includes the CA SiteMinder® session spec

SessionID

Includes the CA SiteMinder® session ID

UserDN

Includes the User DN

Resource

Includes the requested resource

Action

Includes the requested action

RealmOID

Includes the realm OID

ResponseTime

Includes the average response time in milliseconds of the Policy Servers associated with a CA Web Agent or SDK Agent and API application

Note: To output the ResponseTime to a trace log, include the component Agent_Con_Manager along with the data field ResponseTime in the WebAgentTrace.conf file or other file specified in the Policy Server Configuration Object (ACO) and restart the Policy Server. The Agent_Con_Manager component, or Agent API Connection Manager, calculates the ResponseTime each time a response is received from a Policy Server and keeps a running average. To locate the ResponseTime in the trace log, search for [PrintStats].

Trace Message Data Field Filters

To focus on a specific problem, you can narrow the output of the trace log by specifying a filter based on the value of a data field. For example, if you are having problems with an index.html page, you can filter on resources with an html suffix by specifying Resource:==/html in the trace configuration file. Each filter should be on a separate line in the file.

Filters use the following syntax:

data_field:*filter*

The following types of filters are available:

- == (exact match)
- != (does not equal)

The filters use boolean logic as shown in the following examples:

Action:!=get (all actions except get)

Resource:==/html (all resources ending in /html)

Determine the Content of the Trace Log

The WebAgentTrace.conf file determines the content of the trace log. You can control which components and data items appear in your trace log by modifying the settings of the WebAgentTrace.conf file on your web server. The following factors apply when editing the file:

- Entries are case-sensitive.

  When you specify a component, data field, or filter, the values must match exactly the options in the WebAgentTrace.conf file instructions.

- Uncomment the configuration settings lines.
- If you modify the WebAgentTrace.conf file before installing a new agent over an existing agent, the file is overwritten. Rename or back up the file first. After the installation, you can integrate your changes into the new file.

Follow these steps:

1. Open the WebAgentTrace.conf file.

   Note: We recommend duplicating the original file and changing the copy. Modifying the copy preserves the default settings.

2. Add components and subcomponents using the following steps:

   a. Find the section that matches your type of agent. For example, if you have an Apache 2.0 Agent that is installed on your server, look for a line resembling the following example:

      ```
      # For Apache 2.0, Apache 2.2, IIS 7.0 and SunOne Web Agents
      ```

   b. Locate the following line in that section:

      ```
      #components:
      ```

   c. Uncomment the line. Then add the component names that you want after the colon. Separate multiple components commas as shown in the following example:

      ```
      components:  AgentFramework, HTTPAgent
      ```

   d. (Optional) Follow the component name with the name of a subcomponent you want. Separate the subcomponent name with a slash as shown in the following example:

      ```
      components:  AgentFramework/Administration
      ```

3. Add data fields and filters using the following steps:

   a. Locate the following line in the appropriate section:

      ```
      #data:
      ```

   b. Uncomment the line. Then add the data fields that you want after the colon. Separate multiple data fields with commas as shown in the following example:

      ```
      data: Date, Time, Pid, Tid, TransactionID, Function, Message, IPAddr
      ```

   c. (Optional) Add filters to your data fields by following the data field with a colon, the Boolean operator and the value you want. The values you specify for the filters must match exactly. The following example shows a filter which logs activities for a specific IP address:

      ```
      data: Date, Time, Pid, Tid, TransactionID, Function, Message,
      IPAddr:==127.0.0.1
      ```

      Note: Each filter must be on a separate line in the file.

4. Save your changes and close the file.

5. Restart the web server to apply your changes.

   The content of the trace log has been determined.

Limit the Number of Trace Log Files Saved

You can limit the number of trace logs that a CA SiteMinder® agent keeps. For example, if you want to save disk space on the system that stores your agent logs, you can limit the number of trace logs using the following parameter:

TraceFilesToKeep

Specifies the number of CA SiteMinder® agent trace log files that are kept. New trace logs are created in the following situations:

- When the agent starts.
- When the size limit of the trace log (specified by the value of the TraceFileSize parameter) is reached.

Changing the value of this parameter does *not* automatically delete any existing trace logs which exceed the number that you want to keep. For example, If your system has 500 trace logs stored, and you decide to keep only 50 of those files, the agent does *not* delete the other 450 trace logs.

Setting the value of this parameter to zero retains all the trace logs.

Default: 0

Follow these steps:

1. Archive or delete any existing trace logs from your system.
2. Set the value of the TraceAppend parameter to no.
3. Change the value of the TraceFilesToKeep parameter to the number of trace logs that you want to keep.

Collect Detailed Agent Connection Data with an Agent Connection Manager Trace Log

To collect detailed information about the connections between a SiteMinder WSS Agent and Policy Server, you create a Trace Log file that contains information gathered by the Agent Collection Manager.

Follow these steps:

1. Open your Agent Configuration object or local configuration file.
2. Set the value of the TraceFile parameter to yes.

   Note: Setting the value of this parameter to yes in a local configuration file of a web server overrides any of the logging settings defined on the Policy Server. For example, when the value of this parameter is set to yes in a LocalConfig.conf file log files are generated even if the value of the AllowLocalConfig parameter in the corresponding Agent Configuration object on the Policy Server is set to no. Additionally, set the related trace logging parameters (that define the file name, size, and so on) in the LocalConfig.conf file to override any Policy Server trace log settings.
3. Specify the full path to the trace log file for your Agent Connection Data in the TraceFileName parameter. This is the file that contains the trace log output.

4. Set the value of the TraceConfigFile parameter to the full path of the following file:

   agent_home/config/AgentConMgr.conf

   agent_home

   Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

   Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

   Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

   Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

5. Define the format the trace log file for your Agent Connection Data by setting the following parameters:

   TraceAppend

   Adds new logging information to the end of an existing log file instead of rewriting the entire file each time logging is invoked.

   Default: No

   TraceDelimiter

   Specifies a custom character that separates the fields in the trace file.

   Default: No default

   Example: |

   TraceFileSize

   Specifies (in megabytes) the maximum size of a trace file. The Web Agent creates a new file when this limit is reached.

   Default: 0 (a new log file is not created)

   Example: 20 (MB)

   TraceFormat

   Specifies how the trace file displays the messages. Choose *one* of the following options:

   - default—uses square brackets [] to enclose the fields.
   - fixed—uses fields with a fixed width.
   - delim—uses a character of your choice to delimit the fields.
   - xml—uses XML-like tags. A DTD or style sheet is *not* provided with the Web Agent.

   Default: default (square brackets)

LogLocalTime

Specifies whether the logs use Greenwich Mean Time (GMT) or local time. To use GMT, change this setting to no. If this parameter does not exist, the default setting is used.

Default: Yes

6. Restart your web server so the new settings take effect.

   Detailed information about the SiteMinder WSS Agent connections is collected.

   Note: For CA SiteMinder® 12.51, the BusyHandleCount and FreeHandleCount attributes are not used.

Configure XML Message Processing Logging

In addition to Web Agent logging functionality, the SiteMinder WSS Agent provides an additional level of log information relating specifically to its processing of XML messages. SiteMinder WSS Agent logging is implemented using Apache's *log4j* standard (see http://logging.apache.org).

Note: SiteMinder WSS Agent logging does not start until an XML message that needs to be processed is received.

By default, SiteMinder WSS Agent logging is enabled and written to the soasm_agent.log file in:

- Windows—*agent_home*\bin\
- UNIX—*agent_home*/bin/

agent_home

Indicates the directory where the SiteMinder WSS Agent is installed on your web server.

Default (Windows 32-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent

Default (Windows 64-bit SiteMinder WSS Agent installations: C:\Program Files\CA\Web Services Security\webagent\win64

Default (Windows 32-bit SiteMinder WSS Agent installations operating on 64-bit systems: [set the PRF value for your book]\CA\Web Services Security\webagent\win32

You can change logging parameters for your SiteMinder WSS Agent by editing the log.config file, which can be found in:

- Windows—*agent_home*\config\
- UNIX— *agent_home*/config/

Disable SiteMinder WSS Agent XML Message Processing Logging

To disable SiteMinder WSS Agent XML message processing logging, remove or comment out (using a "#" prefix) the following lines from the log.config file located in the Agent config subdirectory:

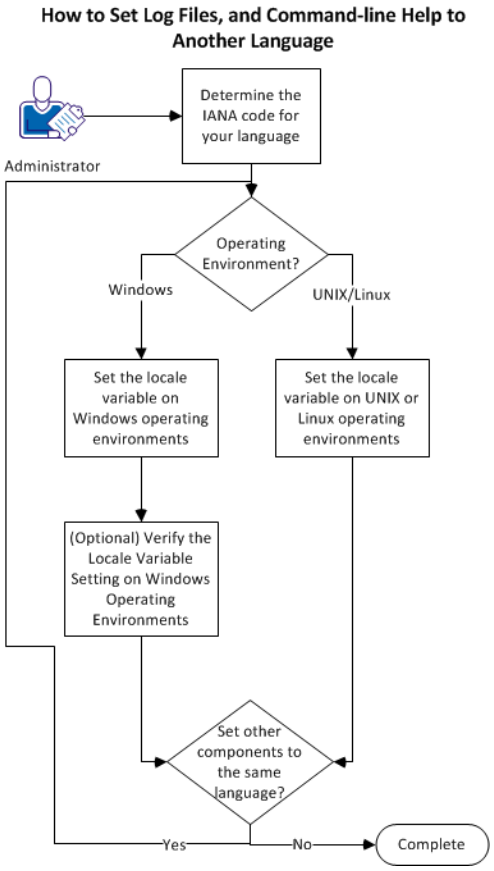
```
log4j.appender.A2=org.apache.log4j.DailyRollingFileAppender
log4j.appender.A2.File=${NETE_TXM_ROOT}/bin/soasm_agent.log
```

Set Log Files, and Command-line Help to Another Language

The following components support log files, and command-line help in other languages:

- The Policy Server
- The Web Agent
- The Report Server
- The CA SiteMinder Agent for SharePoint
- The CA SiteMinder® SPS
- SiteMinder WSS Agents
- Any custom software that is created with the CA SiteMinder® SDK.

The following graphic describes the work flow for setting log files, and command-line help to another language:



Follow these steps:

1. Determine the IANA code for your language (see page 109).
2. Create the environment variable for your operating environment using one of the following procedures:
   - Set the locale variable on Windows operating environments (see page 110).
   - Set the locale variable on UNIX or Linux operating environments (see page 112).
3. (Optional) Verify the locale variable setting on windows operating environments (see page 111).
4. (Optional) Repeat Steps 1 through 3 to set any other components in your environment to the same language.

Determine the IANA Code for Your Language

Each language has a unique code. The Internet Assigned Numbers Authority (IANA) assigns these language codes. Adding a language code to a locale variable changes the language that the software displays. Determine the proper code for the language that you want before creating the locale variable.

The following table lists the IANA codes that correspond to the languages supported by the software:

Language	IANA Code
Brazilian Portuguese	pt_BR
French	fr
German	de
Italian	it
Japanese	ja
Korean	ko
Simplified Chinese	zh-Hans
Spanish	es

Note: A list of IANA language codes is available from this third-party website.

Environment Variables

The environment variables are settings by which users can customize a computer to suit their needs. Examples of environment variables include the following items:

- A default directory for searching or storing downloaded files.
- A username.
- A list of locations to search for executable files (path).

Windows operating environments allow global environment variables, which apply to all users of a computer. The environment variables on UNIX or Linux operating environments must be set for each user or program.

To set the locale variable, pick the procedure for your operating environment from the following list:

- Set the locale variable on Windows operating environments (see page 110).
- Set the locale variable on UNIX or Linux operating environments (see page 112).

Set the Locale Variable on Windows Operating Environments

The following locale variable specifies the language settings for the software:

```
SM_ADMIN_LOCALE
```

Create this variable and set it to the language that you want. Set this variable on *each* component for which you want to use another language. For example, suppose you want to have a Policy Server and an agent that is set to French. Set this variable on both of those components to French.

Note: The installation or configuration programs do *not* set this variable.

Follow these steps:

1. Click Start, Control Panel, System, Advanced system settings.

   The system properties dialog appears.
2. Click the Advanced tab.
3. Click Environment Variables.
4. Locate the System variables section, and then click New.

   The New System Variable dialog opens with the cursor in the Variable name: field.
5. Type the following text:

   ```
   SM_ADMIN_LOCALE
   ```
6. Click the Variable name: field, and then type the IANA language code (see page 109) that you want.
7. Click OK.

   The New System Variable dialog closes and the SM_ADMIN_LOCALE variable appears in the list.
8. Click OK *twice*.

   The locale variable is set.
9. (Optional) Repeat Steps 1 through 8 to set other components to the same language.

Verify the Locale Variable Value on Windows Operating Environments

You can very the value to which the locale variable is set at any time. You can do this procedure after setting the variable to confirm that it is set correctly.

Note: Instructions for verifying the variable value on UNIX and Linux are in the setting procedure (see page 112).

Follow these steps:

1. Open a command-line window with the following steps:
   a. Click Start, Run.
   b. Type the following command:

      ```
      cmd
      ```

   c. Click OK.

      A command-line window opens.
2. Enter the following command:

   ```
   echo %SM_ADMIN_LOCALE%
   ```

   The locale appears on the next line. For example, when the language is set to German, the following code appears:

   ```
   de
   ```

   The value of the locale variable is verified.

Set the Locale Variable on UNIX or Linux Operating Environments

The following locale variable specifies the language settings for the software:

`SM_ADMIN_LOCALE`

Create this variable and set it to the language that you want. Set this variable on *each* component for which you want to use another language. For example, suppose you want to have a Policy Server and an agent that is set to French. Set this variable on both of those components to French.

Note: The installation or configuration programs do *not* set this variable.

Follow these steps:

1. Log in to the computer that is running the component that you want.
2. Open a console (command-line) window.
3. Enter the following command:

   `export SM_ADMIN_LOCALE=IANA_language_code`

   The command in the following example sets the language to French:

   `export SM_ADMIN_LOCALE=fr`

   The locale variable is set.
4. (Optional) Verify that the locale variable is set properly by entering the following command:

   `echo $SM_ADMIN_LOCALE`

   The locale appears on the next line. For example, when the language is set to German, the following code appears:

   `de`
5. (Optional) Repeat Steps 1 through 4 to set other components to the same language.

Chapter 12: Troubleshooting

This section contains the following topics:

Incorrect Error Code Returned Returned on XML-DCC Authentication Failure

Valid on Oracle Directory Enterprise Edition (formerly Oracle iPlanet Directory Server Enterprise Edition)

Symptom:

Authentication against the XML Document Credential Collector authentication scheme fails, but the web server returns a 500 Internal Server Error instead of a 403 Forbidden error.

Solution:

Perform the following steps:

1. Open the obj.conf file on your web server.
2. Locate the following line:

   ```
   AuthTrans fn="SiteMinderAgent"
   ```

3. Add UseOutputStreamSize="0"to the end of the previous line, as shown in the following example:

   ```
   AuthTrans fn="SiteMinderAgent" UseOutputStreamSize="0"
   ```

4. Save the file, and then restart the web server.

Web Server Starts but Web Agent Not Enabled

Symptom:

The Web Agent is not enabled even though the web server has started.

Solution:

Open the WebAgent.conf file, and then set the EnableWebAgent parameter to yes.

smget Error Message When Web Server Starts

Valid on Oracle iPlanet web servers

Symptom:

When starting the Web Server, you see the message:

shmget failed. You may be trying to make a cache that is too large.

Solution:

Make the recommended adjustments to the shared memory segments.

More information:

How to Tune the Solaris 10 Resource Controls (see page 84)

Reconfigured Web Agent Won't Operate

Valid on Oracle iPlanet web servers

Symptom:

Web Agent configuration changes are not in the obj.conf file. The Web Agent cannot operate.

Solution:

The Oracle iPlanet Administration console was used to make server modifications before the changes the Agent configuration program made to the obj.conf were applied. Reconfigure the Web Agent.

Oracle iPlanet Web Server Fails at Runtime

Symptom:

Oracle iPlanet web server is failing at run time.

Solution:

Set the value of the StackSize setting (in the magnus.conf file of the Oracle iPlanet server) to 256 KB. The magnus.conf file is located in:

Oracle_iPlanet_home/*web_server_instance*/config

Appendix A: Worksheets

This section contains the following topics:

Agent Installation Worksheet

Use the following table to record the information that the agent installation program requires:

Information Needed	Your Value
Installation Directory	

Agent Configuration Worksheet

Use the following table to record the information that the agent configuration program requires:

Information Needed	Your Value
Host Registration (Yes/No)	
Admin User Name	
Admin Password	
Enable Shared Secret Rollover	
Trusted Host Name	
Host Configuration Object	
IP Address	
FIPS Mode Setting	
SmHost.conf file Name	
SmHost.conf file Location	
Select Server	
Agent Configuration Object Name	

Information Needed	Your Value
Advanced Authentication Scheme Dialog	