CA SiteMinder®

Web Agent Option Pack Guide

r12.0 SP3

Fourth Edition

CA Technologies Product References

This document references the following CA Technologies products:

- SiteMinder®

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.

Documentation Changes

The following documentation updates have been made since the third edition of this documentation:

- Environment Variables Added by the Installation (see page 11)—Updated to remove the reference to an incorrect .jar file (144569, 147387, 147392).
- Web Agent Option Pack Uninstall and Upgrade information deleted.

Contents

Chapter 1: Introduction

This section contains the following topics:

Features Provided by the Web Agent Option Pack

The Web Agent Option Pack is a stand-alone component that must be installed separately from SiteMinder. It installs Federation Web Services (FWS) on an application server or Web server.

Note: Unlike the Web Agent Option Pack, the Policy Server Option Pack is no longer a stand-alone component but an integral part of the Policy Server and its installation.

The Web Agent Option Pack supports the following SiteMinder features:

Federation Security Services

SAML 1.0, 1.1, 2.0 and WS-Federation

SAML and WS-Federation are standards that define the exchange of authentication and authorization information between entities in a network using assertions. To support these standards, the Web Agent Option Pack is required.

Federation Web Services (FWS)

Federation Web Services is an application that enables production and consumption of SAML assertions. FWS is installed by the Web Agent Option Pack on a server that is connected to the Policy Server.

SAML Affiliate Agent

The SAML Affiliate Agent is a stand-alone component that provides authentication and session management for a consumer site or affiliate that does not have a SiteMinder Policy Server and Web Agent to protect its resources. The Web Agent Option Pack installs Federation Web Services, which allows the SAML Affiliate Agent to communicate with the Policy Server.

Note: For more information about Federation Security Services, see the *Federation Security Services Guide*.

eTelligent Rules

eTelligent Rules are policy expressions that combine Boolean operators and user-defined variables and that are evaluated at runtime. As policy expressions, eTelligent Rules allow administrators to implement fine-grained access control of protected resources on a SiteMinder-protected Web site. To support POST variables, the Web Agent Option Pack is required.

Note: For more information about eTelligent Rules, see the *Policy Server Configuration Guide*.

Chapter 2: Installation Requirements and Considerations

Components Required for Federation Security Services

The following components are required for Federation Security Services:

- Policy Server
- Application Server or Web Server

  Note: An application server with built-in Web server, such as JBOSS, WebLogic or WebSphere, is required to deploy Federation Web Services. Alternately, a Web server with application server plug-in, such as ServletExec, can be used.
- Web Agent Option Pack

Components Required for eTelligent Rules

The following components are required for eTelligent Rules:

- Policy Server
- Web Agent Option Pack

  The Web Agent Option Pack is required to support eTelligent Rules that contain POST variables.

General Option Pack Installation Requirements

Before you can install the Web Agent Option Pack, the following components are required:

- Supported application server

  For the supported ServletExec versions, see the Platform Support Matrix at the Technical Support site.

  Note: Be sure to apply the most current hot fixes for ServletExec. Federation Web Services requires these hot fixes to work with ServletExec. To obtain the hot fixes, go to the New Atlanta Communications web site.

- A supported Java Development Kit (JDK).

  This JDK is required even if you are using an application server that ships with a JDK or JRE.

- For Linux operating platforms, be sure that the required Linux libraries are installed.

Note: You can install the Web Agent Option Pack without the Web Agent. However, install the Web Agent *before* using eTrust SiteMinder FSS.

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 libraries are required on the host system:

Red Hat 5.x

compat–gcc-34-c++-3.4.6-*patch_version*.I386

Red Hat 6.x (32-bit)

libstdc++-4.4.6-3.el6.i686.rpm

To have the appropriate 32-bit C run–time library for your operating environment, install the previous rpm.

Red Hat 6.x (64-bit)

libXau-1.0.5-1.el6.i686.rpm

libxcb-1.5-1.el6.i686.rpm

libstdc++-4.4.6-4.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

Version Compatibility

The version of the Web Agent Option Pack must match the Policy Server version, including the Service Pack and CR version. In addition, if the Web Agent and Web Agent Option Pack are installed on the same machine, they must also be the same version, including the Service Pack and CR version.

Environment Variables Added by the Installation

The installation of the Web Agent Option Pack sets the following environment variables:

- NETE_WA_OPACK = “INSTALLED”
- NETE_WA_PATH = $NETE_WA_ROOT$$/$bin;$NETE_WA_ROOT$$/bin/$thirdparty; $NETE_JRE_ROOT$$/$bin;$NETE_JRE_ROOT$$/bin/$server

Web Agent Option Pack on JBOSS 5.1.2 Requires Workaround to Operate (152769)

Symptom:

On a JBoss 5.1.2 server, system JARs are overriding application-specific JARs, such as those JARs for the Web Agent Option Pack.

Solution:

Prevent the Web Agent Option Pack XML API files from being overwritten by JBOSS system JARS.

Important! This workaround only applies to JBOSS 5.1.2 and higher.

Add the following filter package in two places in the war-deployers-jboss-beans.xml file:

```
<property name="filteredPackages">javax.servlet,org.apache.commons.
logging,javax.xml.parsers,org.xml.sax,org.w3c.dom,javax.xml.namespace</property>
```

The filter package allows the use of the Web Agent Option Pack XML API files instead of the JBOSS system files.

Follow these steps:

1. Locate the war-deployers-jboss-beans.xml file located in the following directory:

   ```
   /deployers/jbossweb.deployer/META-INF/
   ```

2. Find the following entry:

   ```
   <property name="filteredPackages">javax.servlet,org.apache.
   commons.logging</property>
   ```

3. Change the entry to:

   ```
   <property name="filteredPackages">javax.servlet,org.apache.commons.
   logging,javax.xml.parsers,org.xml.sax,org.w3c.dom,javax.xml.namespace</proper
   ty>
   ```

   This entry in the file is on one line.

4. Find the second instance of the entry in step 2 and replace it with the entry in step 3.

   Add the filter package in both places in the XML file.

5. Save the XML file.

Chapter 3: Install the Web Agent Option Pack

This section contains the following topics:

Installation Modes

The Web Agent Option Pack can be installed on a Web server running ServletExec or other supported application server, such as WebLogic, WebSphere, or JBoss.

You can install the Web Agent Option Pack in one of the following modes:

- GUI Mode

  GUI mode uses a graphical installation wizard to install the Web Agent Option Pack.

- Console Mode (UNIX platforms only)

  Uses command line questions about the installation in a UNIX console window.

- Unattended Mode

  Installs the Web Agent Option Pack without user intervention. Use the unattended installation mode to automate additional installations on other machines in your network.

Note: You must install the Web Agent Option Pack using GUI or Console mode *before* running an unattended installation. The initial installation creates a properties file that contains the installation settings for the unattended installation.

Run the Web Agent Option Pack Installer

The Web Agent Option Pack can be installed as a stand-alone product. The installer will try to find an installed Web Agent, but if it cannot, it prompts you to continue or cancel. If you continue, the installer prompts you for an installation path and installs the Option Pack in the location that you specify.

Install the Web Agent Option Pack using using the method for your platform:

- Windows systems: install in GUI mode.

  Important! If you are installing the Web Agent Option Pack on a Windows system immediately after installing the Web Agent, reboot your machine first.

- UNIX systems: install in GUI or console mode.

  To install in console mode, you execute the Option Pack binary with the -i console command argument.

To install the Web Agent Option Pack in GUI or console mode

1. Stop the Web or application server and exit any applications that are running.
2. Log in to the CA Technical Support site.
3. Click Download Center.
4. Search the Download Center for the installation kit you need.
5. Run the installation program according to your platform.
   - **Windows:** Double-click the executable.
   - **UNIX:** At the command prompt, type one of the following commands:
     - **GUI Mode**

       ./binary_filename

     - **Console Mode**

       ./binary_filename -i console

     Example: To run the installation in GUI mode on an AIX platform, enter:
     ./ca-wa-opack-12.0-sp*x*-aix.bin

     x is the number of the appropriate SP.

     Note: If needed, use the chmod command to add execute permissions to the installation file, for example:

     chmod +x ca-wa-opack-12.0-sp*x*-aix.bin
6. Follow the installation dialogs and prompts to complete the installation.
7. When the installation is complete, choose whether to reboot your system now or later; then click Done.

To reinstall the Option Pack, re-run the executable.

Source the Environment Script on a UNIX Operating Environments

After you install the Web Agent Option Pack on a UNIX system, the installation program creates an environment script (ca-wa-opack-env.sh).

Source the environment script so the library path of the application server points to the location of the Web Agent Option Pack /bin directory.

Source the script by entering the following command at the command line:

```
. ./ca-wa-opack-env.sh
```

Setting the correct library path lets the option pack and the web or application server to work together.

After you source the script, the library path is set. The variable name for the library path differs depending on the operating system. Example of several library paths:

Solaris/Linux

LD_LIBRARY_PATH=/*webagent_option_pack_home*/bin

HP-UX

SHLIB_PATH=/*webagent_option_pack_home*/bin

AIX

LIBPATH=/*webagent_option_pack_home*/bin

Important! The application server startup script can reset the library path. Ensure that the path to the Web Agent Option Pack is the first entry in the path.

The path to the Web Agent Option Pack environment script points to one of the following locations:

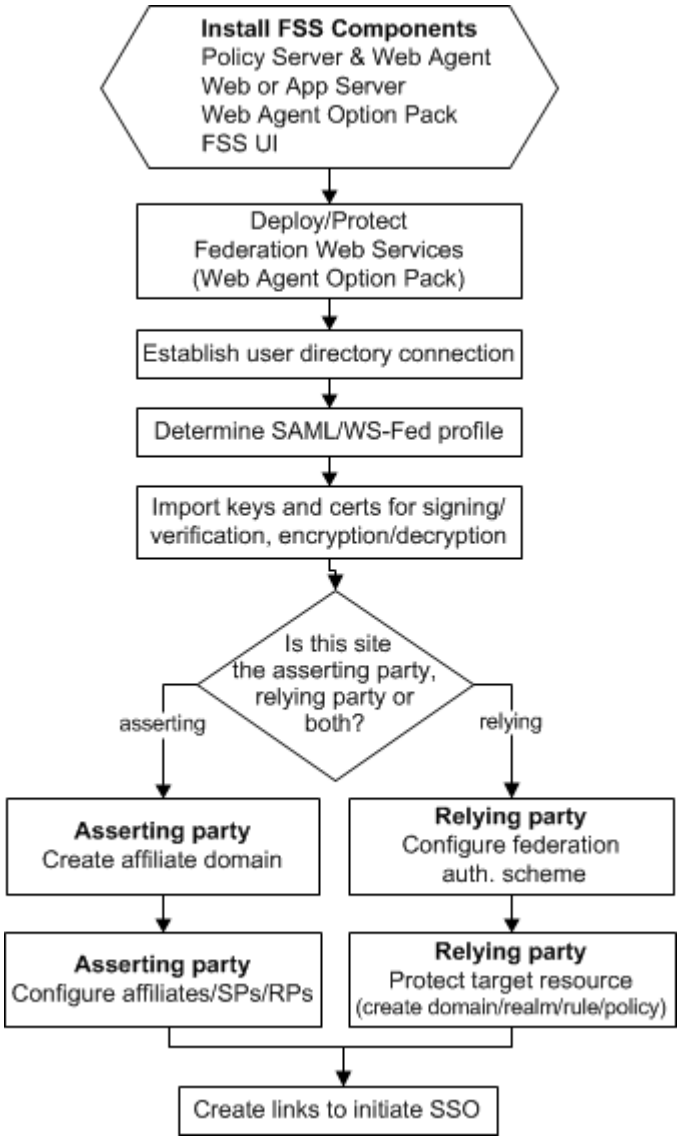
- The installation directory of the web agent option pack. The default location is: /*webagent_option_pack_home*/bin.
- The installation directory of the web agent.

  If you install the option pack on the same system as the web agent, the script resides in the web agent directory. For any UNIX installation, the default location is /*web_agent_home*/bin.

What to do Next

For Federation Security Services, configure the components necessary to establish federated partnerships. Most of these components are configurable using the [set the ufi variable for your book].

The following flow chart highlights the general configuration process.



Chapter 4: Deploy Federation Web Services as a Web Application

This section contains the following topics:

Federation Web Services Application Overview

Federation Web Services (FWS) is a collection of servlets that are packaged as a web application in accordance with the Java Servlet API 2.3 specification. The Federation Web Services application is installed with the Web Agent Option Pack. The application is deployed within an application server, or deployed inside the Tomcat web server, which is embedded in the SPS federation gateway.

The web application is rooted at a specific URL within the web server, http://www.*your_webserver.com*/affwebservices/. URLs for the servlets included with the FWS application have this same root.

The Federation Web Services application provides these services:

- Assertion Retrieval Service (SAML 1.x)
- SAML credential collector (SAML 1.x)
- Intersite Transfer Service (SAML 1.x)
- Artifact Resolution Service (SAML 2.0)
- Assertion Consumer Service (SAML 2.0)
- Security Token Consumer Service (WS-Federation)
- AuthnRequest service (SAML 2.0)
- Single Sign-on service (SAML 2.0 and WS-Federation)
- Single Logout Service (SAML 2.0)
- Signout Service (WS-Federation)

- Session Synchronization—ValidateSession & Logout calls—a value-added service, supporting a standards-based SOAP RPC mechanism (for the SAML Affiliate Agent only).
- Notification Alert—a value-added service, supporting a standards-based SOAP RPC mechanism (for the SAML Affiliate Agent only)

Note: Session synchronization and notification alert services are only used when the SiteMinder SAML Affiliate Agent is at the relying party. These services are not supported with the SPS federation gateway.

Properties File for Federation Web Services

The AffWebServices.properties file contains all the initialization parameters for Federation Web Services. For deploying FWS, set only the parameter that specifies the location of the WebAgent.conf file. For the rest of the settings, accept the default values or modify as needed.

The settings are as follows:

AffWebServices.properties Settings	Value
NotificationLibraryType	Specifies the library type the Web Agent uses for notification alerts. **Note:** The SPS federation gateway does not support this service.
NotificationLibraryDetails	Indicates the Java classname or the C library and function name. **Note:** The SPS federation gateway does not support this service.
SMserverPort	Determines which Policy Server service at the producer processes the notification tunnel calls.
AgentConfigLocation	Indicates the location of the WebAgent.conf file. If you are using a 4.x IIS or Sun ONE Web Agent, this field can be left blank.

The installed location of the AffWebServices.properties file is in the following locations:

- For a web or application server

  web_agent_or_webagent_option_pack_home/affwebservices/WEB-INF/classes

- For the SPS federation gateway:

  sps_home/secure-proxy/Tomcat/webapps/affwebservices/WEB-INF/classes

web_agent_home

Indicates the installed location of the Web Agent.

sps_home

Indicates the installed location of the Secure Proxy Server.

Set up the LoggerConfig.properties File

The LoggerConfig.properties file lets you enable logging so the Federation Web Services application can record the following information:

- Assertion retrieval
- Session management
- Notification alert information
- Trace messages

The log file shows activity at the asserting party and the relying party, depending on how your site is configured.

Note: The LoggerConfig.properties file is in UTF-8 format. If you plan to modify this file, use an editor that supports this format.

The installed location of the LoggerConfig.properties file is:

- For the Web Agent, the location is

  web_agent_home/affwebservices/WEB-INF/classes
- For an application server

  deployment_directory/affwebservices/WEB-INF/classes
- For the SPS federation gateway:

  sps_home/secure-proxy/Tomcat/webapps/affwebservices/WEB-INF/classes

web_agent_home

Indicates the installed location of the Web Agent.

deployment_directory

Indicates the default deployment directory for your application server.

sps_home

Indicates the installed location of CA SiteMinder SPS.

Modify the settings as needed. If a value is not specified, the default value for the default locale is used.

The following table shows the settings in the LoggerConfig.properties file.

LoggerConfig.properties Settings	**Description**
EnableDNSLookup	Instructs the FWS application whether to do a DNS or reverse DNS lookup when processing an incoming SAML request at the consuming site. Select Y or N. When an incoming SAML request is received at a consumer site, FWS logs the details of the request, including the requesting host name. The DNS lookup call collects the host name. The default behavior is to do the DNS lookup. If you select N for this heading, the DNS call is not made and the IP address is logged instead.
LoggingOn (required)	Enables log output. Select Y or N.
LocalFileName (required)	Names the file to use for log output.
LogLocalTime	Enables use of local time for log messages. Select Y or N.

LoggerConfig.properties Settings	Description
LogRollover	Defines the type of rollover functionality. Select Y or N then define the LogSize or LogCount parameter.
LogSize	Specifies the maximum file size, in megabytes, when rolling over log files by size.
LogCount	Specifies how many log output files to leave when roll-over is enabled.
TracingOn	Enables trace log output. Select Y or N.
TraceFileName	Names the file to use for trace log output.
TraceConfig	Specifies the trace configuration file. For more information, see Trace Logging.
TraceRollover	Defines the type of rollover functionality for tracing. Select Y or N and then specify a TraceSize or TraceCount value.
TraceSize	Specifies the maximum file size, in megabytes, when rolling over trace log files by size.
TraceCount	Specifies how many trace log output files to leave when roll-over is enabled.
TraceFormat	Specifies the trace output file format (default, fixed-width fields, delimited format, XML)
TraceDelim	Defines the character to use as a delimiter when using fixed-width fields as the trace format.

Deploy Federation Web Services as a Web Application

If you are using the Web Agent Option Pack, deploy the Federation Web Services (FWS) application into operation.

Configure one of the following application servers to work with FWS:

- Set Up ServletExec to Work with Federation Web Services.
- Set Up WebLogic to Work with Federation Web Services (see page 28).

- Set Up WebSphere to Work with Federation Web Services (see page 34).
- Set Up a JBOSS or Tomcat to Work with Federation Web Services.

If you are using the SPS federation gateway, Federation Web Services is already deployed on the embedded Tomcat server.

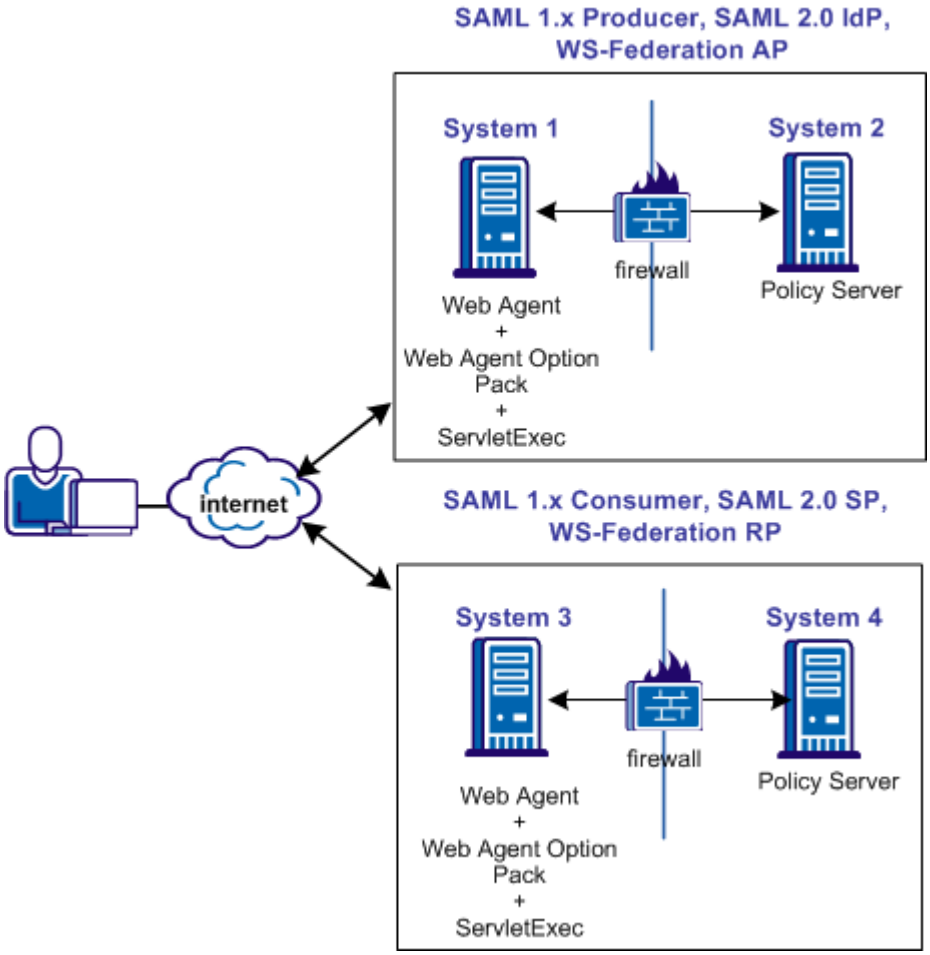
Configure ServletExec to Work with Federation Web Services

For the Federation Web Services application to work with ServletExec, deploy Federation Web Services as a web application for ServletExec at the asserting and relying party.

Note: SiteMinder r12.0 SP3 is shipped with a ServletExec license key file named ServletExec_AS_6_license_key.txt. If you do not have this license key, contact CA Technical Support. From this license file, copy the license key and enter it in the ServletExec License dialog of the ServletExec Administration Console. For instructions on licensing ServletExec, see ServletExec documentation, available at the New Atlanta Communication http://www.newatlanta.com website.

Important! If ServletExec runs in the context of your web server, such as ServletExec ISAPI or NSAPI, install the option pack on the same system where the SiteMinder Web Agent is installed. If ServletExec does run in the context of your web server, such as ServletExec AS, you can install the Web Agent Option Pack on a different system than the SiteMinder Web Agent.

The following illustration shows a SiteMinder and ServletExec sample configuration, where ServletExec, the Web Agent Option Pack, and the Web Agent are installed on the same server.



Important! Apply the most current hot fixes for ServletExec. Without the hot fixes, Federation Web Services does not work with ServletExec. To obtain the hot fixes, go to the New Atlanta Communication website.

To set up ServletExec to work with FWS

1. Open the ServletExec Administration Console.
2. Under Web Applications, select manage.

   The Manage Web Applications dialog opens.
3. Click Add a Web Applications.

4. Enter the following information:
   a. Application Name: affwebservices
   b. URL Context Path: /affwebservices/
   c. Location: affwebservices_home

      Example:

      C:\program files\ca\webagent\affwebservices
5. Click Submit.
6. Exit the ServletExec Console.

Source the Environment Script on a UNIX Operating Environments

After you install the Web Agent Option Pack on a UNIX system, the installation program creates an environment script (ca-wa-opack-env.sh).

Source the environment script so the library path of the application server points to the location of the Web Agent Option Pack /bin directory.

Source the script by entering the following command at the command line:

```
. ./ca-wa-opack-env.sh
```

Setting the correct library path lets the option pack and the web or application server to work together.

After you source the script, the library path is set. The variable name for the library path differs depending on the operating system. Example of several library paths:

Solaris/Linux

LD_LIBRARY_PATH=/*webagent_option_pack_home*/bin

HP-UX

SHLIB_PATH=/*webagent_option_pack_home*/bin

AIX

LIBPATH=/*webagent_option_pack_home*/bin

Important! The application server startup script can reset the library path. Ensure that the path to the Web Agent Option Pack is the first entry in the path.

The path to the Web Agent Option Pack environment script points to one of the following locations:

- The installation directory of the web agent option pack. The default location is: */webagent_option_pack_home*/bin.
- The installation directory of the web agent.

  If you install the option pack on the same system as the web agent, the script resides in the web agent directory. For any UNIX installation, the default location is */web_agent_home*/bin.

Modify the FWS Properties File for a ServletExec Deployment

The AffWebServices.properties file contains all the initialization parameters for Federation Web Services. For deploying FWS, set only the parameter that specifies the location of the WebAgent.conf file.

To configure the AffWebServices.properties file

1. Navigate to the AffWebServices.properties file. For ServletExec, go to *web_agent_home*/affwebservices/WEB-INF/classes.
2. Set the AgentConfigLocation parameter to the location of the WebAgent.conf file at each partner site.
   - Windows example:

     C:\\Program Files\\ca\\webagent\\bin\\IIS\\WebAgent.conf

     Note: Federation Web Services is a Java component, so the Windows paths must contain double backslashes.
   - UNIX example:

     server_home/servers/https-hostname/config/WebAgent.conf
   - Windows example for the SPS federation gateway

     sps_home\\proxy-engine\\conf\\defaultagent\\WebAgent.conf
   - UNIX example for the SPS federation gateway

     sps_home/proxy-engine/conf/defaultagent/WebAgent.conf
3. Repeat this procedure for each application server where the Web Agent Option Pack is installed.
4. Accept the default values for the rest of the settings.

Enable ServletExec to Write to the IIS File System

The IIS server user account must have proper rights for IIS to allow a plug-in to write to its file system. For ServletExec to write to the federation log files, the anonymous user account that is associated with ServletExec must have permissions to write to the file system.

Follow these steps:

1. Open the IIS Internet Information Services Manager on the system where ServletExec is installed.
2. Navigate to Web Sites, Default Web Site.

   The set of applications is displayed in the right pane.
3. Select ServletExec and right-click Properties.
4. Select the Directory Security tab in the Properties dialog.
5. Click Edit in the Authentication and access control section.

   The Authentication Methods dialog opens.
6. Set the controls as follows.
   a. Select Enable Anonymous Access.

      For anonymous access, enter a name and password of a user account that has the permissions to right to the Windows file system. To grant this right to a user account, see Windows documentation. For example, you can use the IUSR Internet Guest account for anonymous access.
   b. Clear Basic authentication.
   c. Clear Integrated Windows authentication.
7. If prompted, apply the security changes to all child components of the web server.
8. Restart the web server.

The user account that is associated with ServletExec can now write to the IIS file system.

Follow these steps:

1. Open Control Panel, Administrative Tools, Local Security Policy, Local Policies, User Rights Assignment.

   The Local Security Settings dialog displays.
2. Double-click Act as part of the operating system.

   The Act as part of the operating system Properties dialog opens.
3. Add the anonymous user account to the Local Security Setting dialog.
4. Click OK.
5. Exit from the control panel.

Optionally, we strongly recommend that you look at the Agent Configuration Object for the Web Agent protecting the IIS Web Server. This object verifies that the SetRemoteUser parameter is set to yes to preventing any anonymous user from writing to the file system.

Ensure the IIS Default Web Site Exists

The Web Agent requires the IIS Web Server to have a Default Web Site for proper installation. The Default Web Site is automatically installed with the IIS Web Server. If this website does not exist, install the SiteMinder virtual directories to a different website on IIS. To install the SiteMinder virtual directories to a different website on IIS, edit the Metabase.

A technical note on the site describes the Technical Support site changes that are needed. To find the note:

1. Go to the main Support page.
2. Select Literature, Tech Notes.
3. Select the document titled METABASE -3 Error.

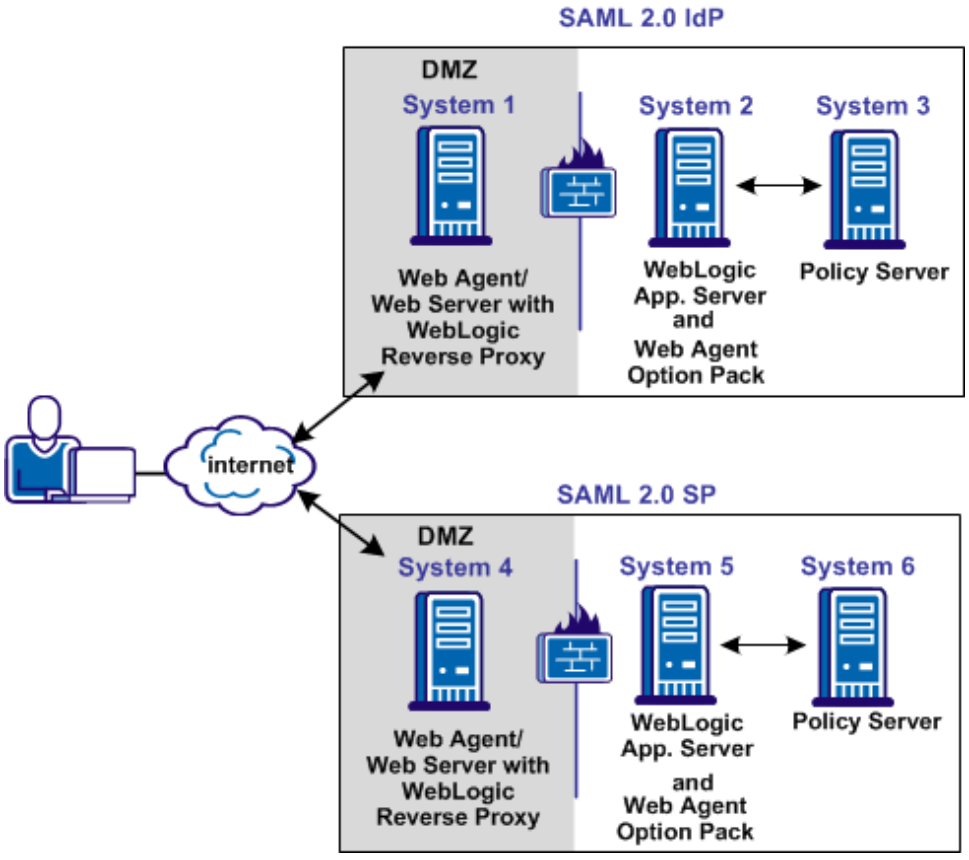
   The documents are listed in alphabetical order.

Set Up WebLogic to Work with Federation Web Services

To enable Federation Web Services (FWS) for a SiteMinder/WebLogic configuration, deploy the FWS application.

Note: For a list of supported version of WebLogic, see the SiteMinder r12.0 SP3 Platform Support Matrix on the Technical Support site.

The following illustration shows a SiteMinder and WebLogic sample configuration. The illustration provides an example of how to use FWS in a sample federated environment.



In this environment, deploy the FWS application on System 2 and System 5.

Important! Complete the deployment procedure for the Web Agent at the asserting party and the relying party.

After installing the software components on the systems in the illustration, deploy the FWS application. Deploy the application on System 2 for the asserting party and on System 5 for the relying party.

To deply the FWS application

1. Set the LD_LIBRARY_PATH Variable
2. Create a SmHost.conf File
3. Create a WebAgent.conf File
4. Modify the AffWebServices.properties File
5. Configure the WebLogic Reverse Proxy Plug-in.
6. Deploy the FWS Application on WebLogic.

   Important! For the FWS application to work with WebLogic Server, review the weblogic.xml file in the WEB-INF directory. Verify that the prefer-web-inf-classes parameter in the weblogic.xml file is set to true.

   For instructions on reviewing the weblogic.xml file, go to Deploy the FWS Application on WebLogic (see page 33).

Source the Environment Script on a UNIX Operating Environments

After you install the Web Agent Option Pack on a UNIX system, the installation program creates an environment script (ca-wa-opack-env.sh).

Source the environment script so the library path of the application server points to the location of the Web Agent Option Pack /bin directory.

Source the script by entering the following command at the command line:

```
. ./ca-wa-opack-env.sh
```

Setting the correct library path lets the option pack and the web or application server to work together.

After you source the script, the library path is set. The variable name for the library path differs depending on the operating system. Example of several library paths:

Solaris/Linux

LD_LIBRARY_PATH=/*webagent_option_pack_home*/bin

HP-UX

SHLIB_PATH=/*webagent_option_pack_home*/bin

AIX

LIBPATH=/*webagent_option_pack_home*/bin

Important! The application server startup script can reset the library path. Ensure that the path to the Web Agent Option Pack is the first entry in the path.

The path to the Web Agent Option Pack environment script points to one of the following locations:

- The installation directory of the web agent option pack. The default location is: /*webagent_option_pack_home*/bin.
- The installation directory of the web agent.

  If you install the option pack on the same system as the web agent, the script resides in the web agent directory. For any UNIX installation, the default location is /*web_agent_home*/bin.

Create an SmHost.conf File

The FWS application requires an SmHost.conf file. However, the Web Agent Option Pack does not install this file, so you must create it.

To create an SmHost.conf

1. Go to the directory /*webagent_option_pack_home*/bin
2. Run the smreghost.exe.

   For instructions on running smreghost.exe, see the *Web Agent Installation Guide*.
3. Put the SmHost.conf file in the following directory on Systems 2 and 5:

   /*webagent_option_pack_home*/config

Create a WebAgent.conf File

The FWS application requires the WebAgent.conf file. However, the Web Agent Option Pack does not install this file, so you must create it.

To create a WebAgent.conf file

1. Copy the WebAgent.conf file from System 1 to the following directory on System 2 and System 5:

   /webagent_option_pack_home/config

   webagent_option_pack_home

   Defines the installed location of the Web Agent Option Pack on System 2 or System 5.

2. Modify the WebAgent.conf file by:

   a. Setting the EnableWebAgent parameter to YES.

   b. Modifying other configuration parameters to suit FWS.

The following sample shows a WebAgent.conf file for the FWS application:

```
# WebAgent.conf - configuration file for the Federation Web Services Application
#agentname="agent_name, IP_address"
HostConfigFile="/webagent_option_pack/config/SmHost.conf"
AgentConfigObject="agent_config_object_name"
EnableWebAgent="YES"
```

Note: The Agent Configuration Object referenced in this WebAgent.conf file must be a new object that you create. Do not specify the object in use by the Web Agent installed in your environment.

Modify the AffWebServices.properties File for WebLogic

The AffWebServices.properties file contains all the initialization parameters for Federation Web Services. For deploying FWS, set only the parameter that specifies the location of the WebAgent.conf file.

To configure the AffWebServices.properties file

1. Navigate to the AffWebServices.properties file. For ServletExec, go to *web_agent_home*/affwebservices/WEB-INF/classes.
2. Set the AgentConfigLocation parameter to the location of the WebAgent.conf file at each partner site.
   - Windows example:

     C:\\Program Files\\ca\\webagent\\bin\\IIS\\WebAgent.conf

     Note: Federation Web Services is a Java component, so the Windows paths must contain double backslashes.
   - UNIX example:

     server_home/servers/https-hostname/config/WebAgent.conf
   - Windows example for the SPS federation gateway

     sps_home\\proxy-engine\\conf\\defaultagent\\WebAgent.conf
   - UNIX example for the SPS federation gateway

     sps_home/proxy-engine/conf/defaultagent/WebAgent.conf
3. Repeat this procedure for each application server where the Web Agent Option Pack is installed.

Accept the default values for the rest of the settings.

Configure the WebLogic Reverse Proxy Plug-in

To set up the WebLogic Reverse Proxy plug-in:

1. On System 1, configure the WebLogic reverse proxy plug-in on the Apache Web Server.

   For more information, see WebLogic documentation.

2. Add the following aliases to the configuration file of the web server.

   This example uses the Apache httpd.conf file.

```
<IfModule mod_weblogic.c>
WebLogicHost <WebLogic_Machine_IP Address>
WebLogicPort <WebLogic_Machine_Port_Number>
</IfModule>

<Location /affwebservices>
SetHandler weblogic-handler
Debug ALL
</Location>
```

Deploy the FWS Application on WebLogic

Deploy the FWS application on System 2 and System 5.

Important! For the FWS application to work with WebLogic Server, review the weblogic.xml file in the WEB-INF directory. Verify that the prefer-web-inf-classes parameter is set to true.

The weblogic.xml file is located in the directory webagent\affwebservices\WEB-INF.

The following code excerpt shows how to set the prefer-web-inf-classes parameter:

```
<weblogic-web-app>
 <container-descriptor>
  <prefer-web-inf-classes>true</prefer-web-inf-classes>
 </container-descriptor>
</weblogic-web-app>
```

In addition, verify that the precompile parameter is set to true, as listed following:

```
<jsp-descriptor>
   <precompile>true</precompile>
<jsp-descriptor>
```

Follow these steps: to deploy FWS

1. Use the WebLogic Server Console and deploy FWS. The FWS application is installed in:

   /webagent_option_pack_home/affwebservices/

   For more information about deploying a web application, see the WebLogic documentation.

2. Test that the FWS application is working. Open a web browser and enter:

   http://*fqhn*:*port_number*/affwebservices/assertionretriever

   fqhn

   Defines the fully qualified host name.

   port_number

   Defines the port number of the server where the Federation Web Services application is installed.

   For example:

   http://myhost.ca.com:81/affwebservices/assertionretriever

   If Federation Web Services is operating correctly, you see the following message:

   ```
   Assertion Retrieval Service has been successfully initialized.
   The requested servlet accepts only HTTP POST requests.
   ```

   This message indicates that Federation Web Services is listening for data activity. The FWS application is now deployed for the WebLogic server.

   If Federation Web Services is not operating correctly, a message that the Assertion Retrieval Service has failed displays. If the service fails, review the Federation Web Services log.

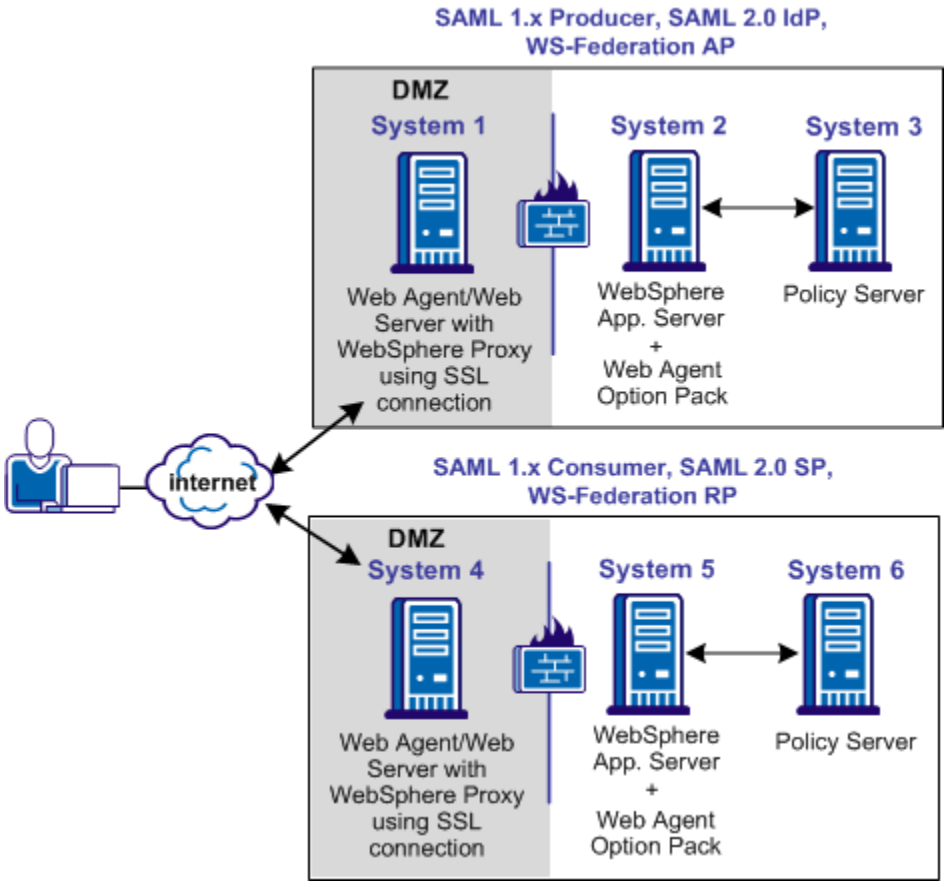
Note: For instructions on enabling trace logging for the FWS application, see Trace Logging.

Set Up WebSphere to Work with Federation Web Services

To enable FWS in a federated environment for a SiteMinder/WebSphere Application Server (WAS) configuration, deploy the FWS application.

On Systems 2 and 5, deploy FWS. These systems must also have WAS and the associated WAS Fix Pack, if applicable. On Systems 1 and 4, install the Web Agent and the WAS Proxy Plug-in. Enable SSL between the proxy and the WAS.

The following illustration shows a SiteMinder and WebSphere sample configuration.



Prerequisites:

- Install WAS on systems that have the WebSphere Application Server installed.
- Complete the deployment procedure for the Web Agent at the asserting party and the relying party.

After installing the software components on the systems in the illustration, deploy FWS on System 2 and System 5 by following these steps:

1. Set the WebSphere LD_LIBRARY_PATH variable.
2. Create a SmHost.conf file.
3. Create a WebAgent.conf file.
4. Modify the AffWebServices.properties file.
5. Copy option pack library files to WebSphere.
6. Deploy a Federation Web Services WAR File in WebSphere.

Source the Environment Script on a UNIX Operating Environments

After you install the Web Agent Option Pack on a UNIX system, the installation program creates an environment script (ca-wa-opack-env.sh).

Source the environment script so the library path of the application server points to the location of the Web Agent Option Pack /bin directory.

Source the script by entering the following command at the command line:

```
. ./ca-wa-opack-env.sh
```

Setting the correct library path lets the option pack and the web or application server to work together.

After you source the script, the library path is set. The variable name for the library path differs depending on the operating system. Example of several library paths:

Solaris/Linux

LD_LIBRARY_PATH=/*webagent_option_pack_home*/bin

HP-UX

SHLIB_PATH=/*webagent_option_pack_home*/bin

AIX

LIBPATH=/*webagent_option_pack_home*/bin

Important! The application server startup script can reset the library path. Ensure that the path to the Web Agent Option Pack is the first entry in the path.

The path to the Web Agent Option Pack environment script points to one of the following locations:

- The installation directory of the web agent option pack. The default location is: /*webagent_option_pack_home*/bin.
- The installation directory of the web agent.

  If you install the option pack on the same system as the web agent, the script resides in the web agent directory. For any UNIX installation, the default location is /*web_agent_home*/bin.

Create an SmHost.conf File

The FWS application requires the SmHost.conf file. However, the Web Agent Option Pack does not install this file, so you must create it.

To create an SmHost.conf file

1. Create an SmHost.conf file by running smreghost.exe, which is located in the following directory:

   /*webagent_option_pack_home*/bin

2. Put the SmHost.conf file in the following directory on System 2 and System 5:

   /*webagent_option_pack_home*/config

Create a WebAgent.conf File

The FWS application requires the WebAgent.conf file; however, the Web Agent Option Pack does not install this file so you must create it.

To create a WebAgent.conf file

1. Copy the WebAgent.conf file from System 1 to the following directory on System 2 and System 5:

   /*webagent_option_pack_home*/config

   where,

   webagent_option_pack_home

   Defines the installed location of the Web Agent Option Pack on System 2 and System 5.

2. Modify the WebAgent.conf file by:

   a. Setting the EnableWebAgent parameter to YES.

   b. Modifying any other configuration parameters to suit the environment for the FWS application.

The following sample shows a WebAgent.conf file for the FWS application:

```
# WebAgent.conf - configuration file for the Federation Web Services Application
#agentname="<agent_name>, <IP_address>"
HostConfigFile="/<webagent_option_pack>/config/SmHost.conf"
AgentConfigObject="<agent_config_object_name>"
EnableWebAgent="YES"
```

Note: The Agent Configuration Object referenced in this WebAgent.conf file must be a new object that you create. Do not specify the object in use by the Web Agent installed in your environment.

Modify the AffWebServices.properties File for WebSphere

The AffWebServices.properties file contains all the initialization parameters for Federation Web Services. For deploying FWS, set only the parameter that specifies the location of the WebAgent.conf file.

To configure the AffWebServices.properties file

1. Navigate to the AffWebServices.properties file. For ServletExec, go to *web_agent_home*/affwebservices/WEB-INF/classes.
2. Set the AgentConfigLocation parameter to the location of the WebAgent.conf file at each partner site.
   - Windows example:

     C:\\Program Files\\ca\\webagent\\bin\\IIS\\WebAgent.conf

     Note: Federation Web Services is a Java component, so the Windows paths must contain double backslashes.
   - UNIX example:

     server_home/servers/https-hostname/config/WebAgent.conf
   - Windows example for the SPS federation gateway

     sps_home\\proxy-engine\\conf\\defaultagent\\WebAgent.conf
   - UNIX example for the SPS federation gateway

     sps_home/proxy-engine/conf/defaultagent/WebAgent.conf
3. Repeat this procedure for each application server where the Web Agent Option Pack is installed.

Accept the default values for the rest of the settings.

Copy Web Agent Option Pack Libraries to WebSphere

To copy the Web Agent Option Pack library files on System 2 and System 5

1. Copy the following files from the directory *webagent_option_pack*\bin
   - smcommonutil.dll
   - smerrlog.dll
   - smfedclientcomponent.dll
   - smjavaagentapi.dll
   - libetpki_openssl_ssl.a (AIX only)
   - libetpki_openssl_crypto.a (AIX only)
2. Place the copied libraries in the following directory:

   WebSphere_home\AppServer\bin

Deploy a Federation Web Services WAR File in WebSphere

To deploy the FWS WAR file

1. Create a WAR file of the Federation Web Services application. The application is installed in:

   webagent_option_pack\affwebservices\

   For more information about creating a WAR file, see WebSphere documentation.

2. Deploy the WAR file using WebSphere Administrator Console.

   For more information, see WebSphere documentation.

   Important! If you make subsequent changes to any of the properties files in the affwebservices directory, recreate a WAR file and redeploy this file in the application server.

3. From the WebSphere Administrator Console, go to Applications, Enterprise Applications.

4. Select the name of the web services WAR file, such as affwebservices_war.

5. On the Configuration tab:

   a. Set the Classloader Mode.

      There are two possible modes for class loading:

      - Classes loaded with the parent class loader first (default)
      - Classes loaded with the local class loader first

      The mode you select is implementation-dependent. In releases before 7.0, these modes were named PARENT_FIRST and PARENT_LAST. See the WebSphere documentation for further information.

   b. Set WAR Classloader Policy to Application.

   c. Save the settings.

6. Test that the Federation Web Services application is working by opening a web browser and entering:

   http://*fqhn*:*port_number*/affwebservices/assertionretriever

   where,

   fqhn

   Defines the fully qualified host name.

   port_number

   Defines the port number of the server where the Federation Web Services application is installed.

For example:

http://myhost.ca.com:81/affwebservices/assertionretriever

If Federation Web Services is operating correctly, the following message appears:

```
Assertion Retrieval Service has been successfully initialized.
The requested servlet accepts only HTTP POST requests.
```

This message indicates that Federation Web Services is listening for data activity.

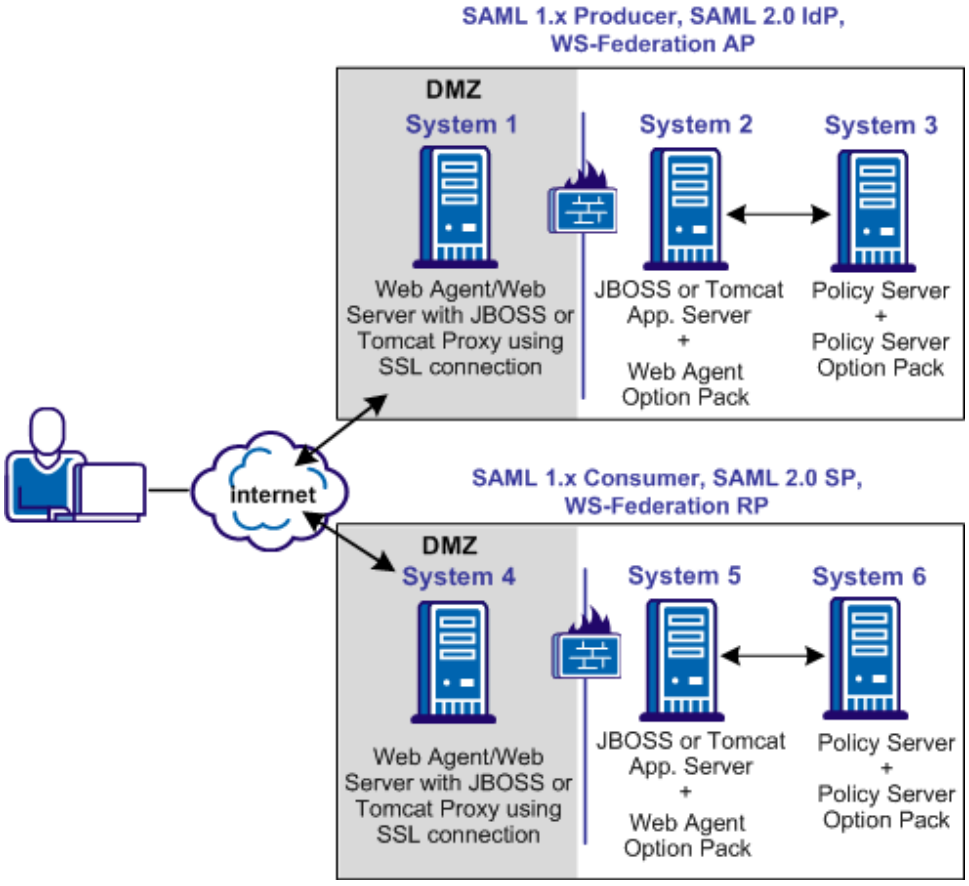
When the Federation Web Services is not operating correctly, a message states that the Assertion Retrieval Service has failed. If the Assertion Retrieval Service fails, verify the Federation Web Services log.

Note: For more information about enabling trace logging for the FWS application, see Trace Logging.

Configure JBOSS or Tomcat to Work with Federation Web Services

To use a JBoss or Tomcat Application Server in a SiteMinder federated environment, deploy the FWS application on the application server.

The following illustration shows the deployment with JBOSS or Tomcat. On Systems 1 and 4, the Web Agent is installed with the JBOSS or Tomcat Connector for proxy support. SSL is enabled between the proxy and the application server. On Systems 2 and 5, FWS is deployed with the application server by way of the Web Agent Option Pack.



The process for deploying FWS is as follows:

1. Create an SmHost.conf file.
2. Create a WebAgent.conf file.
3. Modify the AffWebServices properties file.
4. Deploy the FWS WAR file in the application server.
5. Modify the Tomcat catalina.properties file (Tomcat 6.0.18 and higher).

The following sections detail each step in the process.

Source the Environment Script on a UNIX Operating Environments

After you install the Web Agent Option Pack on a UNIX system, the installation program creates an environment script (ca-wa-opack-env.sh).

Source the environment script so the library path of the application server points to the location of the Web Agent Option Pack /bin directory.

Source the script by entering the following command at the command line:

```
. ./ca-wa-opack-env.sh
```

Setting the correct library path lets the option pack and the web or application server to work together.

After you source the script, the library path is set. The variable name for the library path differs depending on the operating system. Example of several library paths:

Solaris/Linux

LD_LIBRARY_PATH=/*webagent_option_pack_home*/bin

HP-UX

SHLIB_PATH=/*webagent_option_pack_home*/bin

AIX

LIBPATH=/*webagent_option_pack_home*/bin

Important! The application server startup script can reset the library path. Ensure that the path to the Web Agent Option Pack is the first entry in the path.

The path to the Web Agent Option Pack environment script points to one of the following locations:

- The installation directory of the web agent option pack. The default location is: /*webagent_option_pack_home*/bin.
- The installation directory of the web agent.

  If you install the option pack on the same system as the web agent, the script resides in the web agent directory. For any UNIX installation, the default location is /*web_agent_home*/bin.

Create an SmHost.conf File

The FWS application requires the SmHost.conf file. However, the Web Agent Option Pack does not install this file, so you must create it.

1. Create an SmHost.conf file by running smreghost.exe, which is located in the following directory:

   /webagent_option_pack/bin

2. Put the SmHost.conf file in the following directory on System 2 and System 5:

   /webagent_option_pack/config

Create a WebAgent.conf File

The FWS application requires the WebAgent.conf file. However, the Web Agent Option Pack does not install this file, so you must create it.

1. Copy the WebAgent.conf file from System 1 to the following directory on System 2 and System 5:

   /webagent_option_pack/config

   where,

   webagent_option_pack

   Defines the installed location of the Web Agent Option Pack on System 2 and System 5.

2. Modify the WebAgent.conf file as follows:

   a. Set the EnableWebAgent parameter to YES.

   b. Modify any other configuration parameters to suit the environment for the FWS application.

   The following sample shows a WebAgent.conf file for the FWS application:

```
# WebAgent.conf - configuration file for the Federation Web Services Application
#agentname="agent_name, IP_address"
HostConfigFile="/web_agent_home/config/SmHost.conf"
AgentConfigObject="agent_config_object_name"
EnableWebAgent="YES"
```

Note: The Agent Configuration Object referenced in this WebAgent.conf file must be a new object that you create. Do not specify the object in use by the Web Agent installed in your environment.

Modify the AffWebServices.properties File for JBOSS or Tomcat

The AffWebServices.properties file contains all the initialization parameters for Federation Web Services. For deploying FWS, set only the parameter that specifies the location of the WebAgent.conf file.

To configure the AffWebServices.properties file

1. Navigate to the AffWebServices.properties file. For ServletExec, go to *web_agent_home*/affwebservices/WEB-INF/classes.
2. Set the AgentConfigLocation parameter to the location of the WebAgent.conf file at each partner site.
   - Windows example:

     C:\\Program Files\\ca\\webagent\\bin\\IIS\\WebAgent.conf

     Note: Federation Web Services is a Java component, so the Windows paths must contain double backslashes.
   - UNIX example:

     server_home/servers/https-hostname/config/WebAgent.conf
   - Windows example for the SPS federation gateway

     sps_home\\proxy-engine\\conf\\defaultagent\\WebAgent.conf
   - UNIX example for the SPS federation gateway

     sps_home/proxy-engine/conf/defaultagent/WebAgent.conf
3. Repeat this procedure for each application server where the Web Agent Option Pack is installed.

Accept the default values for the rest of the settings.

Deploy an FWS WAR File in JBoss or Tomcat

To deploy the FWS application for a JBOSS or Tomcat server

1. Open a command window and navigate to the affwebservices directory, which is located in:

   /webagent_option_pack/affwebservices/.
2. Create a WAR file by entering the command:

   ```
   jar cvf affwebservices.war *
   ```

   For more information about deploying a Web application, see the documentation for your application server.

3. Copy the affwebservices.war file to the appropriate server location:

   JBOSS

   JBOSS_home/server/default/deploy/

   JBOSS_home is the installed location of the JBOSS application server.

   Important! For JBOSS, deploy affwebservices in an exploded state. Refer to https://access.redhat.com/knowledge/solutions/34813.

   Tomcat

   Tomcat_home/webapps

   Tomcat_home is the installed location of the Tomcat application server.

4. Restart the application server.

5. After the server has restarted, access the JBOSS or Tomcat Administrative Console. All the services that affwebservices supports appear on the main Console page.

6. Test that the FWS application is working by opening a web browser and entering the following URL:

   http://*fqhn*:*port_number*/affwebservices/assertionretriever

   fqhn

   Represents the fully qualified host name and

   port_number

   Specifies the port number of the server where the Federation Web Services application is installed.

   For example:

   http://myhost.ca.com:81/affwebservices/assertionretriever

   If FWS is operating correctly, the following message is displayed:

   ```
   Assertion Retrieval Service has been successfully initialized.
   The requested servlet accepts only HTTP POST requests.
   ```

   This message indicates that FWS is listening for data activity. The FWS application is now deployed for the application server.

   When FWS is not operating correctly, a message states that the Assertion Retrieval Service has failed. If there is a failure, review the FWS log.

   Note: For more information about enabling trace logging for the FWS application, see Trace Logging.

Modify the Tomcat catalina.properties File (Tomcat 6.0.18 or higher)

If you deploy Federation Web Services on an Apache Tomcat server 6.0.18 or higher, a configuration change for SAML POST and Artifact transactions is required. Add a configuration parameter to the catalina.properties file.

Follow these steps:

1. Navigate to *Tomcat_home*/conf.

   Tomcat_home

   Specifies the installed location of the Tomcat application server.

2. Open the catalina.properties file.

3. Add the following configuration parameter in the "String cache configuration" section of the file:

   org.apache.catalina.STRICT_SERVLET_COMPLIANCE=true

   The following sample shows the entries in the Catalina.properties file:

```
# String cache configuration.
tomcat.util.buf.StringCache.byte.enabled=true
#tomcat.util.buf.StringCache.char.enabled=true
#tomcat.util.buf.StringCache.trainThreshold=500000
#tomcat.util.buf.StringCache.cacheSize=5000
org.apache.catalina.STRICT_SERVLET_COMPLIANCE=true
```

For Apache Tomcat 6.0.32 or later, add the following configuration parameter to the "String cache configuration" section of the catalina.properties file. This file exists in the *Tomcat_home*/conf. directory. *Tomcat_home* is the installed location of the Tomcat application server.

```
org.apache.tomcat.util.http.ServerCookie.ALLOW_EQUALS_IN_VALUE=true
```

The following sample shows the entries in the "String cache configuration" section of the Catalina.properties file:

```
# String cache configuration.
tomcat.util.buf.StringCache.byte.enabled=true
#tomcat.util.buf.StringCache.char.enabled=true
#tomcat.util.buf.StringCache.trainThreshold=500000
#tomcat.util.buf.StringCache.cacheSize=5000
org.apache.catalina.STRICT_SERVLET_COMPLIANCE=true
org.apache.tomcat.util.http.ServerCookie.ALLOW_EQUALS_IN_VALUE=true
```

Chapter 5: Unattended Mode Installation

This section contains the following topics:

How to Run an Unattended Mode Installation

After the Web Agent Option Pack is installed on one system, you can automate installations on other Web or application servers using the Web Agent Option Pack's unattended mode installation. An unattended mode installation lets you install or uninstall the Web Agent Option Pack without any user intervention.

To run an unattended mode installation, you must:

1. Run an installation in GUI or Console mode to install the unattended installation properties file.
2. Modify the properties file.
3. Run the unattended installation.

This chapter covers the following topics:

Prepare an Unattended Mode Installation

An unattended mode installation uses the ca-wa-opack-installer.properties file to propagate the Option Pack installation set-up across all servers in your network. You can define the installation parameters in the properties file then copy the properties file and the Web Agent Option Pack executable file to any applicable server in your network. After the files are copied, you can run an unattended installation.

To prepare an unattended mode installation

1. Run an initial installation of the Web Agent Option Pack in GUI or Console mode.

   This installation will install the ca-wa-opack-installer.properties file.

2. Open the ca-wa-opack-installer.properties file, and if needed, modify the settings.

   The properties file is in the directory *web_agent_opack_home*/install_config_info.

   USER_INSTALL_DIR

   Specifies the Web Agent Option Pack's installation location.

   USER_REQUESTED_RESTART

   Specifies restarting the machine after installation.

   Note: These default values were saved in the properties file during the initial installation.

3. Save the properties file.

Run an Unattended Mode Installation

After completing the preparation steps, you can run an unattended mode installation.

To run an unattended mode installation

1. From a system where the Web Agent Option Pack is already installed, copy the following files to a local directory on the system where you want to install the Option Pack.
   - Web Agent Option Pack executable or binary
     - ca-wa-opack-12.0-sp2-*platform*.exe
     - ca-wa-opack-12.0-sp2-*operating_system*.bin
   - ca-wa-opack-installer.properties
2. Open a console window and navigate to the location where you copied the files.
3. Execute the following command:

   agent_executable -f *properties_file* -i silent

   Windows example:

   ca-wa-opack-12.0-sp2-win32.exe -f ca-wa-opack-installer.properties -i silent

   Solaris example:

   ./ca-wa-opack-12.0-sp2-sol.bin -f ca-wa-opack-installer.properties -i silent

   Note: These examples assume that you are running the installation from the directory containing the executable and properties files. If you are not running the installation from this directory, specify the full path to these files. If there are spaces in the directory path, enclose the entire path in quotation marks.

A status box opens and shows the progress of the unattended installation. When the installation is complete, the command prompt is redisplayed.

4. Determine that the installation completed successfully by viewing the log file CA_SiteMinder_Option_Pack_12.0-sp2_for_Web_Agent_ InstallLog.log. This file is located in the directory *web_agent_opack_home*/install_config_info.

Note: You can run an unattended mode installation that reinstalls the Web Agent Option Pack on the original system where it was installed in GUI or console mode.

Stop an Unattended Mode Installation in Progress

To manually stop an unattended mode installation in progress, follow the instructions for your platform:

Windows

Open the Windows Task Manager, and stop the following two processes:

- ca-wa-opack-12.0-sp3-win32.exe
- wa_option_pack.exe

UNIX

Type Ctrl+C.

Appendix A: International Support

An *internationalized* product is an English product that runs correctly on local language versions of the required operating system and required third-party products and that supports local language data for input and output. Internationalized products also support the ability to specify local language conventions for date, time, currency, and number formats.

A *translated* product (sometimes referred to as a *localized* product) is an internationalized product that includes local language support for the product's user interface, online help and other documentation, as well as local language default settings for date, time, currency, and number formats.

In addition to the English release of this product, CA SiteMinder r12.0 SP3 Web Access Manager supports *only* those languages listed in the following table.

Important! If you run the product in a language environment *not* listed in the table, you can experience problems.

Language	Internationalized	Translated
Brazilian-Portuguese	Yes	No
Chinese (Simplified)	Yes	No
Chinese (Traditional)	Yes	No
Czech	Yes	No
Danish	Yes	No
Dutch	Yes	No
Finnish	Yes	No
French	Yes	No
German	Yes	No
Greek	Yes	No
Hungarian	Yes	No
Italian	Yes	No
Japanese	Yes	No
Korean	Yes	No
Norwegian	Yes	No
Polish	Yes	No

Language	Internationalized	Translated
Russian	Yes	No
Spanish	Yes	No
Swedish	Yes	No
Turkish	Yes	No

Index

V

W