Sun OpenSSO Enterprise Policy Agent 3.0 Guide for Apache Tomcat 6.0

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

Part No: 820–7251–20
May 14, 2009

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun^{TM} Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

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

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certaines composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la legislation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la legislation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement designés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Sun OpenSSO Enterprise Policy Agent 3.0 Guide for Apache Tomcat 6.0

Last updated: May 14, 2009

The Tomcat 6.0 version 3.0 policy agent is a Java EE agent (formerly called a J2EE agent) that functions with Sun™ OpenSSO Enterprise to protect resources on Apache Tomcat 6.0.

Contents

Note – Sun also provides a version 2.2 policy agent for Tomcat 6.0. However, to use the new version 3.0 policy agent features, you must deploy the Tomcat 6.0 version 3.0 agent described in this guide. For general information about version 3.0 Java EE agents, including the new features, see the *Sun OpenSSO Enterprise Policy Agent 3.0 User's Guide for J2EE Agents*.

Supported Platforms and Web Containers for the Tomcat 6.0 Version 3.0 Agent

Supported Versions of the Tomcat 6.0 Web Container

The Tomcat 6.0 version 3.0 agent is supported on Tomcat 6.0.x releases.

For information about Tomcat 6.0, see `http://tomcat.apache.org/`.

Note – If you plan to use web services security (WSS) and JAX-WS with the Tomcat 6.0 version 3.0 agent, you will need to download and install specific JAX-WS JAR files into the Tomcat 6.0 web container. See "Configuring Web Services Security for the Tomcat 6.0 Version 3.0 Agent" on page 30.

Supported Platforms for the Tomcat 6.0 Version 3.0 Agent

The Tomcat 6.0 version 3.0 agent is supported on these platforms:

- Solaris OS on SPARC platforms, versions 9 and 10 (32-bit and 64-bit platforms)
- Solaris OS on x86 platforms, versions 9 and 10 (32-bit and 64-bit platforms)
- Red Hat Enterprise Linux Advanced Server 4.0 and 5.0 (32-bit and 64-bit platforms)
- Windows 2003, Enterprise Edition (32-bit and 64-bit platforms)
- Windows 2008, Standard Edition (32-bit and 64-bit platforms)
- Ubuntu 8.x

Compatibility and Coexistence for the Tomcat 6.0 Version 3.0 Agent

Compatibility With Access Manager 7.1 and Access Manager 7 2005Q4

Sun Java System Access Manager 7.1 and Sun Java System Access Manager 7 2005Q4 are compatible with version 3.0 policy agents. However, because Access Manager does not support centralized agent configuration, a version 3.0 agent deployed with Access Manager must store its configuration data locally in the `OpenSSOAgentConfiguration.properties` and `OpenSSOAgentBootstrap.properties` files.

For both configurations, the `OpenSSOAgentBootstrap.properties` file on the server where the agent is deployed contains the information required for the agent to start and initialize itself.

Coexistence With Version 2.2 Policy Agents

OpenSSO Enterprise supports both version 3.0 and version 2.2 agents in the same deployment. The version 2.2 agents, however, must continue to store their configuration data locally in their respective `AMAgent.properties` file. Because the version 2.2 agent configuration data is local to the agent, OpenSSO Enterprise centralized agent configuration is not supported for version 2.2 agents. To configure a version 2.2 agent, you must continue to edit the agent's `AMAgent.properties` file.

The OpenSSO Enterprise Console allows you to create and configure a version 2.2 agent profile under Access Control, *realm-name*, Agents, 2.2 Agents.

For information about version 2.2 agents, see the following documentation collection:

`http://docs.sun.com/coll/1322.1`

Pre-Installation Tasks for the Tomcat 6.0 Version 3.0 Agent

- "Setting Your `JAVA_HOME` Environment Variable" on page 5
- "Downloading and Unzipping the `tomcat_v6_agent_3.zip` Distribution File" on page 6
- "Creating a Password File" on page 7
- "Installing the Tomcat 6.0 Scripts on Windows Systems" on page 7
- "Creating an Agent Administrator" on page 8

Setting Your `JAVA_HOME` Environment Variable

Version 3.0 policy agents, including the `agentadmin` program, require JDK 1.5 or later on the server where you plan to install the agent. Before you install the Tomcat 6.0 version 3.0 agent, set your `JAVA_HOME` environment variable to point to the JDK installation directory.

Downloading and Unzipping the `tomcat_v6_agent_3.zip` Distribution File

▼ To Download and Unzip the `tomcat_v6_agent_3.zip` Distribution File

1. **Login to the server where you want to install the agent.**

2. **Create a directory to unzip the `tomcat_v6_agent_3.zip` distribution file.**

   This guide uses *Agent-HomeDirectory* to represent the directory where you unzip the distribution file.

3. **Download the `tomcat_v6_agent_3.zip` distribution file from one of the following sites:**

   - Sun Downloads site under Identity Management > Policy Agents: `http://www.sun.com/download/index.jsp`
   - OpenSSO project site: `https://opensso.dev.java.net/public/use/index.html`

   The following table shows the files and directories after you unzip the agent distribution file, which are in the following directory:

   Agent-HomeDirectory`/j2ee_agents/tomcat_v6_agent`

   Agent-HomeDirectory is where you unzipped the agent distribution file.

File or Directory	Description
`README.txt` and `license.txt`	Readme and license files
`/bin`	`agentadmin` and `agentadmin.bat` programs
`/config`	Template, properties, and XML files
`/data`	`license.log` file. Do not edit this file.
`/etc`	Agent application (`agentapp.war`) and related file. For information, see "Deploying the Agent Application" on page 21.
`/installer-logs`	Log files written by the `agentadmin` or `agentadmin.bat` program: ■ `/audit` contains local audit trail for the agent instance. ■ `/debug` contains the debug files for the agent instance when the agent runs in debug mode.
`/lib`	Required JAR files
`/locale`	Required properties files

File or Directory	Description
/sampleapp	Policy agent sample application. For information, see "Deploying the Java EE Policy Agent Sample Application" on page 29.

Creating a Password File

A password file is an ASCII text file with only one line specifying the password in clear text. By using a password file, you are not forced to expose a password at the command line during the agent installation. When you install the Tomcat 6.0 version 3.0 agent using the agentadmin program, you are prompted to specify paths to following password files:

- An **agent profile password file** is required for both the agentadmin default and custom installation options.
- An **agent administrator password file** is required only if you use the custom installation option and have the agentadmin program automatically create the agent profile in OpenSSO Enterprise server during the installation.

▼ To Create a Password File

1. **Create an ASCII text file for the agent profile. For example:** /tmp/tomcat6agentpw
2. **If you want the** agentadmin **program to automatically create the agent profile in OpenSSO Enterprise server during the installation, create another password file for the agent administrator. For example:** /tmp/agentadminpw
3. **Using a text editor, enter the appropriate password in clear text on the first line in each file.**
4. **Secure each password file appropriately, depending on the requirements for your deployment.**

Installing the Tomcat 6.0 Scripts on Windows Systems

The Tomcat 6.0 installation file for Windows (.exe extension) does not install certain scripts and related files required by the Tomcat 6.0 version 3.0 agent. Therefore, after you install the Tomcat 6.0 web container on a Windows system, you must copy the scripts from a Tomcat 6.0 .zip distribution file.

▼ To Install the Tomcat 6.0 Scripts on Windows

1. **In a directory separate from the Tomcat 6.0** .exe **installation, download the Tomcat 6.0** .zip **distribution file from** http://tomcat.apache.org/**.**

   For example, download apache-tomcat-6.0.18.zip.

2 **Make sure that the `CATALINA_HOME` environment variable is set to your Tomcat 6.0 `.exe` installation.**

3 **Unzip the Tomcat 6.0 `.zip` distribution file.**

4 **Copy the following files from the unzipped `bin` directory to the Tomcat 6.0 `bin` directory (`${CATALINA_HOME}\bin`):**

- All `.bat` scripts
- `catalina-tasks.xml`
- `.jar` files

Creating an Agent Administrator

An agent administrator can manage agents in OpenSSO Enterprise, including:

- **Agent management**: Use the agent administrator to manage agents either in the OpenSSO Enterprise Console or by executing the `ssoadm` utility.
- **Agent installation**: If you install the agent using the custom installation option (`agentadmin --custom-install`) and want to have the installation program create the agent profile, specify the agent administrator (and password file) when you are prompted.

▼ To Create an Agent Administrator

1 **Login to OpenSSO Enterprise Administration Console.**

2 **Create a new agents administrator group:**

a. **Click Access Control, *realm-name*, Subjects, and then Group.**

b. **Click New.**

c. **In ID, enter the name of the group. For example:** `agentadmingroup`

d. **Click OK.**

3 **Create a new agent administrator user and add the agent administrator user to the agents administrator group:**

a. **Click Access Control, *realm-name*, Subjects, and then User.**

b. **Click New and provide the following values:**

- **ID**: Name of the agent administrator. For example: `agentadminuser`

This is the name you will use to login to the OpenSSO Enterprise Console .

- **First Name** (optional), **Last Name**, and **Full Name**.

  For simplicity, use the same name for each of these values that you specified for ID.
- **Password** (and confirmation)
- **User Status**: Active

c. **Click OK.**

d. **Click the new agent administrator name.**

e. **On the Edit User page, click Group.**

f. **Add the agents administrator group from Available to Selected.**

g. **Click Save.**

4 **Assign read and write access to the agents administrator group:**

a. **Click Access Control,** *realm-name***, Privileges and then on the new agents administrator group link.**

b. **Check "Read and write access to all configured Agents".**

c. **Click Save.**

Next Steps Login into the OpenSSO Enterprise Console as the new agent administrator. The only available top-level tab is Access Control. Under *realm-name*, you will see only the Agents tab and sub tabs.

Installing the Tomcat 6.0 Version 3.0 Agent

Gathering Information to Install the Tomcat 6.0 Version 3.0 Agent

The following table describes the information you will need to provide when you run the agentadmin program to install the Tomcat 6.0 version 3.0 agent. For some agentadmin prompts, you can accept the default value displayed by the program, if you prefer.

TABLE 1 Information Required to Install the Tomcat 6.0 version 3.0 Agent

Prompt	Description
Tomcat Server Config Directory Path	Path to the configuration directory for the Tomcat 6.0 instance. Applies to both default and custom installation options. For example: `/opt/apache-tomcat-6.0.18/conf`
OpenSSO server URL	OpenSSO Enterprise server URL, including the deployment URI. Applies to both default and custom installation options. For example: `https://openssohost.example.com:8080/opensso`
`$CATALINA_HOME` environment variable	Path to the root directory where Tomcat 6.0 is installed. For example: `/opt/apache-tomcat-6.0.18/`
Install policy agent in global `web.xml` file	Option to install the agent filter in the global `web.xml` file: ■ `true` (default): The agent filter is added to the global `web xml` file (`$CATALINA_HOME/conf/web.xml`). Then, every request is intercepted by the agent, so the policy definition needs to reflect the Tomcat 6.0 root URL to access the home page. ■ `false`: The agent filter is not added to the application-specific `web.xml` file. In both cases, agent filter are added to the manager and host manager and applications. See also "Adding Absolute URIs to the Tomcat 6.0 Version 3.0 Agent Profile" on page 19. Applies to the default installation option.

TABLE 1 Information Required to Install the Tomcat 6.0 version 3.0 Agent *(Continued)*

Prompt	Description
Agent URL	Agent URL, including the deployment URI, for the agent application. Applies to both default and custom installation options. For example: `https://agenthost.example.com:8090/agentapp` The `agentapp` is a housekeeping application used by the agent for notifications and other functions such as cross domain single sign-on (CDSSO) support. For more information, see "Deploying the Agent Application" on page 21.
Encryption Key	Key used to encrypt the agent profile password. The encryption key should be at least 12 characters long. You can accept the default key or create a new key using the `agentadmin --getEncryptKey` command. Applies only to the custom installation option.
Agent profile name	A policy agent communicates with OpenSSO Enterprise using the name and password in the agent profile. Applies to both default and custom installation options. For information, see "Creating an Agent Profile" on page 18.
Agent profile password file name	Path to the agent profile password file, which is ASCII text file with only one line specifying the agent profile password. You create the agent profile password file as a pre-installation step. Applies to both default and custom installation options. For information, see "Creating a Password File" on page 7.
Option to the create the agent profile The `agentadmin` program displays the following prompt if the agent profile previously specified for the Agent Profile Name prompt does not already exist in OpenSSO Enterprise: `Enter true if the Agent Profile is being created into OpenSSO by the installer. Enter false if it will be not be created by installer.`	To have the installation program create the agent profile, enter `true`. The program then prompts you for: ■ Agent administrator who can create, update, or delete the agent profile. For example: `agentadmin` **Important**: To use this option, the agent administrator must already exist in OpenSSO Enterprise server. For information see, "Creating an Agent Administrator" on page 8. If you prefer, you can specify `amadmin` as this user. ■ Path to the agent administrator password file. For information, see "Creating a Password File" on page 7. Applies only to the custom installation option.

Installing the Tomcat 6.0 Version 3.0 Agent Using the `agentadmin` Program

The version 3.0 `agentadmin` program includes these installation options:

- Default install (`agentadmin --install`): The program asks a limited number of questions and uses default values for the other options. Use the default install option when the default options, as shown in Table 1, meet your deployment requirements.

  or
- Custom install (`agentadmin --custom-install`): The program asks a full set of questions similar to the version 2.2 program. Use the custom install option when you want to specify values other than the default options shown in Table 1.

Before you install the Tomcat 6.0 version 3.0 agent:

- An OpenSSO Enterprise server instance must be installed and running. To check the server, specify the server URL. For example: `http://opensso-host.example.com:8080/opensso`
- A Tomcat 6.0 server instance must be installed and configured on the machine where you plan to install the agent. For information, see `http://tomcat.apache.org/`.
- You must have downloaded and unzipped the distribution file, as described in "Downloading and Unzipping the `tomcat_v6_agent_3.zip` Distribution File" on page 6.

▼ To Install the Tomcat 6.0 Version 3.0 Agent Using the `agentadmin` Program

1. **Login into the server where you want to install the agent.**

   Important: To install the agent, you must have write permission to the Tomcat 6.0 instance files and directories.

2. **If necessary, shut down the Tomcat 6.0 instance.**

3. **Change to the following directory:**

   PolicyAgent-base/`bin`

4. **On Solaris and Linux systems, set the permissions for the `agentadmin` program as follows, if needed:**

   ```
   # chmod 755 agentadmin
   ```

5. **Start the agent installation:**

   Default install: `# ./agentadmin --install`

   or

Custom install: `# ./agentadmin --custom-install`

On Windows systems, run the `agentadmin.bat` program.

6 **Enter information as requested by the `agentadmin` program, or accept the default values displayed by the program.**

After you have made your choices, the `agentadmin` program displays a summary of your responses. For example, for a custom installation:

```
-----------------------------------------------
SUMMARY OF YOUR RESPONSES
-----------------------------------------------
Tomcat Server Config Directory : /opt/apache-tomcat-6.0.18/conf
$CATALINA_HOME environment variable : /opt/apache-tomcat-6.0.18
OpenSSO server URL : http://opensso-host.example.com:8080/opensso
Agent URL : http://agent-host.example.com:8090/agentapp
Encryption Key : oyFk4DYaNB2kc6MeJ2xnK4hbWtFhabsZ
Agent Profile name : Tomcat6AgentProfile
Agent Profile Password file name : /tmp/tomcat6agentpw
Agent Profile will be created right now by agent installer : true
Agent Administrator : amadmin
Agent Administrator’s password file name : /opt/amadminpw
```

7 **Verify your choices and either continue with the installation (selection 1, the default) , or make any necessary changes.**

If you continue, the program installs the agent and displays a summary of the installation. For example, for a custom installation:

```
SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Bootstrap file location:
/opt/agents/j2ee_agents/tomcat_v6_agent/
   Agent_001/config/OpenSSOAgentBootstrap.properties
Agent Configuration file location
/opt/agents/j2ee_agents/tomcat_v6_agent/
   Agent_001/config/OpenSSOAgentConfiguration.properties
Agent Audit directory location:
/opt/agents/j2ee_agents/tomcat_v6_agent/Agent_001/logs/audit
Agent Debug directory location:
/opt/agents/j2ee_agents/tomcat_v6_agent/Agent_001/logs/debug

Install log file location:
/opt/agents/j2ee_agents/tomcat_v6_agent/installer-logs/audit/custom.log
```

8 **After the installation finishes successfully, if you wish, check the installation logs in the following directory:**

`installer-logs/audit`

9 Restart the Tomcat 6.0 instance that is being protected by the agent.

Note – After you install the Tomcat 6.0 version 3.0 agent for a specific domain, you cannot use that same agent on the same host for a different domain. To use the Tomcat 6.0 version 3.0 agent for another domain on the same host, you must install the agent specifically for that domain.

Example 1 Sample `agentadmin` Program Installation for the Tomcat 6.0 Version 3.0 Agent

```
************************************************************************
Welcome to the OpenSSO Policy Agent for Apache Tomcat 6.0 Servlet/JSP
Container
************************************************************************
Enter the complete path to the directory which is used by Tomcat Server to
store its configuration Files. This directory uniquely identifies the
Tomcat Server instance that is secured by this Agent.
[ ? : Help, ! : Exit ]
Enter the Tomcat Server Config Directory Path
[/opt/apache-tomcat-6.0.18/conf]:

$CATALINA_HOME environment variable is the root of the tomcat
installation.
[ ? : Help, < : Back, ! : Exit ]
Enter the $CATALINA_HOME environment variable: /opt/apache-tomcat-6.0.18

Enter the URL where the OpenSSO server is running. Please include the
deployment URI also as shown below:
(http://opensso.sample.com:58080/opensso)
[ ? : Help, < : Back, ! : Exit ]
OpenSSO server URL: http://opensso-host.example.com:8080/opensso

Enter the Agent URL. Please include the deployment URI also as shown below:
(http://agent1.sample.com:1234/agentapp)
[ ? : Help, < : Back, ! : Exit ]
Agent URL: http://agent-host.example.com:8090/agentapp

Enter a valid Encryption Key.
[ ? : Help, < : Back, ! : Exit ]
Enter the Encryption Key [oyFk4DYaNB2kc6MeJ2xnK4hbWtFhabsZ]:

Enter the Agent profile name
[ ? : Help, < : Back, ! : Exit ]
Enter the Agent Profile name: Tomcat6AgentProfile

Enter the path to a file that contains the password to be used for identifying
the Agent.
[ ? : Help, < : Back, ! : Exit ]
```

```
Enter the path to the password file: /tmp/tomcat6agentpw

WARNING:
Agent profile/User: tomcat30-agent-custom does not exist in OpenSSO
server! Either "Hit the Back button, and re-enter the correct agent profile
name/user name", or "Create this agent profile when asked(available only in
custom-install)", or "Continue without validating it because agent
profile is in sub realm", or "Continue without validating/creating it, and
manually validate/create it in OpenSSO server after installation".

Enter true if the Agent Profile is being created into OpenSSO server by the
installer. Enter false if it will be not be created by installer.
[ ? : Help, < : Back, ! : Exit ]
This Agent Profile does not exist in OpenSSO server, will it be created by the
installer? (Agent Administrator’s name and password are required) [true]:

Agent Administrator is the Administrator user that can create, delete or
update agent profile.
[ ? : Help, < : Back, ! : Exit ]
Enter the Agent Administrator’s name: amadmin

Enter the path to a file that contains the password of Agent Administrator
[ ? : Help, < : Back, ! : Exit ]
Enter the path to the password file that contains the password of Agent
Administrator: /opt/amadminpw

-----------------------------------------------
SUMMARY OF YOUR RESPONSES
-----------------------------------------------
Tomcat Server Config Directory : /opt/apache-tomcat-6.0.18/conf
$CATALINA_HOME environment variable : /opt/apache-tomcat-6.0.18
OpenSSO server URL : http://opensso-host.example.com:8080/opensso
Agent URL : http://agent-host.example.com:8090/agentapp
Encryption Key : oyFk4DYaNB2kc6MeJ2xnK4hbWtFhabsZ
Agent Profile name : Tomcat6AgentProfile
Agent Profile Password file name : /tmp/tomcat6agentpw
Agent Profile will be created right now by agent installer : true
Agent Administrator : amadmin
Agent Administrator’s password file name : /opt/amadminpw
Verify your settings above and decide from the choices below.
1. Continue with Installation
2. Back to the last interaction
3. Start Over
4. Exit
Please make your selection [1]:
Updating the /opt/apache-tomcat-6.0.18/bin/setclasspath.sh script
with the Agent classpath ...DONE.
Creating directory layout and configuring Agent file for Agent_001
```

```
instance ...DONE.
Reading data from file /tmp/tomcat6agentpw and encrypting it ...DONE.
Generating audit log file name ...DONE.
Creating tag swapped OpenSSOAgentBootstrap.properties file for instance
Agent_001 ...DONE.
Creating a backup for file /opt/apache-tomcat-6.0.18/conf/server.xml
...DONE.
Creating a backup for file /opt/apache-tomcat-6.0.18/conf/web.xml ...DONE.
Adding OpenSSO Tomcat Agent Realm to Server XML file :
/opt/apache-tomcat-6.0.18/conf/server.xml ...DONE.
Adding filter to Global deployment descriptor file :
/opt/apache-tomcat-6.0.18/conf/web.xml ...DONE.
Adding OpenSSO Tomcat Agent Filter and Form login authentication to
selected Web applications ...DONE.
SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Bootstrap file location:
/opt/agents/j2ee_agents/tomcat_v6_agent/
  Agent_001/config/OpenSSOAgentBootstrap.properties
Agent Configuration file location
/opt/agents/j2ee_agents/tomcat_v6_agent/
  Agent_001/config/OpenSSOAgentConfiguration.properties
Agent Audit directory location:
/opt/agents/j2ee_agents/tomcat_v6_agent/Agent_001/logs/audit
Agent Debug directory location:
/opt/agents/j2ee_agents/tomcat_v6_agent/Agent_001/logs/debug

Install log file location:
/opt/agents/j2ee_agents/tomcat_v6_agent/installer-logs/audit/custom.log
Thank you for using OpenSSO Policy Agent
```

After You Finish the Install

Agent Instance Directory

The installation program creates the following directory for each agent instance:

PolicyAgent-base/`Agent_`*nnn*

- *PolicyAgent-base* is *Agent-HomeDirectory*/`j2ee_agents/tomcat_v6_agent`, where *Agent-HomeDirectory* is where you unzipped the agent distribution file.
- *nnn* identifies the agent instance as `Agent_001`, `Agent_002`, and so on for each additional agent instance.

Each agent instance directory contains the following subdirectories:

- `/config` contains the configuration files for the agent instance, including `OpenSSOAgentBootstrap.properties` and `OpenSSOAgentConfiguration.properties`.

- `/installer-logs` contains the following subdirectories
  - `/audit` contains local audit trail for the agent instance.
  - `/debug` contains the debug files for the agent instance when the agent runs in debug mode.

Considering Specific Deployment Scenarios for the Tomcat 6.0 Version 3.0 Agent

- "Installing the Tomcat 6.0 Version 3.0 Agent on Multiple Tomcat 6.0 Instances" on page 17
- "Installing the Tomcat 6.0 Version 3.0 Agent on the OpenSSO Enterprise Host Machine" on page 17

Installing the Tomcat 6.0 Version 3.0 Agent on Multiple Tomcat 6.0 Instances

You can install the Tomcat 6.0 version 3.0 agent on multiple Tomcat 6.0 instances on the same host machine. However, you must run the `agentadmin` program for each Tomcat 6.0 instance. During each installation, specify the unique server configuration directory and instance name, so the agent can differentiate the different instances.

Installing the Tomcat 6.0 Version 3.0 Agent on the OpenSSO Enterprise Host Machine

You can install the Tomcat 6.0 version 3.0 agent on a different web container instance on the same host machine where OpenSSO Enterprise server is installed, as long as the web container is supported for both the Tomcat 6.0 version 3.0 agent and OpenSSO Enterprise server.

Required Post-Installation Tasks for the Tomcat 6.0 Version 3.0 Agent

- "Creating an Agent Profile" on page 18
- "Adding Absolute URIs to the Tomcat 6.0 Version 3.0 Agent Profile" on page 19
- "Creating the `manager` and `admin` Groups" on page 20
- "Deploying the Agent Application" on page 21
- "Configuring Tomcat Applications Protected by the Tomcat 6.0 Version 3.0 Agent" on page 21

Creating an Agent Profile

If you created the agent profile using the `agentadmin` program, continue with "Adding Absolute URIs to the Tomcat 6.0 Version 3.0 Agent Profile" on page 19.

The Tomcat 6.0 version 3.0 agent uses an agent profile to communicate with OpenSSO Enterprise server. You can create an agent profile using any of these three methods:

- Allow the `agentadmin` program to create the agent profile during installation when you run the `--custom-install` option. The program prompts you for this information:
  - Agent profile name and path to the agent profile password file
  - Agent administrator name and path to the agent administrator password file
- Use the OpenSSO Enterprise Console.
- Use the `ssoadm` command-line utility with the `create-agent` subcommand. For more information about the `ssoadm` command, see the *Sun OpenSSO Enterprise 8.0 Administration Reference*.

▼ To Create an Agent Profile in the OpenSSO Enterprise Console

1. **Login into the OpenSSO Enterprise Administration Console as `amAdmin`.**

2. **Click Access Control, *realm-name*, Agents, and then J2EE.**

3. **Under Agent, click New.**

4. **In the Name field, enter the name for the new agent profile. For example:**

   `Tomcat6AgentProfile`

5. **Enter and confirm the Password.**

   Important: This password must be the same password that you enter in the agent profile password file that you specify when you run the `agentadmin` program to install the agent.

6. **In the Server URL field, enter the OpenSSO Enterprise server URL.**

   For example: `http://openssohost.example.com:8080/opensso`

7. **In the Agent URL field, enter the URL for the agent application ().**

   For example: `http://agenthost.example.com:8090/agentapp`

   The `agentapp` is a housekeeping application used by the agent for notifications and other functions such as cross domain single sign-on (CDSSO) support. For more information, see "Deploying the Agent Application" on page 21.

8 **Click Create.**

The console creates the agent profile and displays the J2EE Agent page again with a link to the new agent profile, `Tomcat6AgentProfile`.

9 **Click the link to the new agent profile.**

10 **For Login Form URI, add the following entries:**

```
/manager/AMLogin.html
/host-manager/AMLogin.html
```

This step allows the agent to protect the manger and host-manager by default.

11 **Click Save.**

This change (`com.sun.identity.agents.config.login.form` property) is hot-swappable, so you do not need to restart the OpenSSO Enterprise web container for these values to take effect.

Tip – Make a note of the values you specified for the agent profile, including the profile name, password, server URL, and agent URL. You will need these values when you install the Tomcat 6.0 version 3.0 agent using the `agentadmin` program.

Adding Absolute URIs to the Tomcat 6.0 Version 3.0 Agent Profile

---

Note – If you performed this task when you created the agent profile, you can skip it here.

▼ To Add Absolute URIs to the Tomcat 6.0 Version 3.0 Agent Profile

1 **Log in to the OpenSSO Enterprise Administration Console.**

2 **Click Access Control, Top Level Realm, Agents, J2EE, and then the agent profile for theTomcat 6.0 version 3.0 agent.**

3 **On the Edit page, click Application and then Login Processing.**

4 **For Login Form URI, add the following entries:**

```
/manager/AMLogin.html
/host-manager/AMLogin.html
```

5 **Click Save.**

This change (`com.sun.identity.agents.config.login.form` property) is hot-swappable, so you do not need to restart the OpenSSO Enterprise web container for these values to take effect.

Creating the `manager` and `admin` Groups

▼ To Create the `host-manager` and `admin` Groups

1 **Login to OpenSSO Enterprise Administration Console.**

2 **Create two new groups: `manager` and `admin`, as follows:**

a. **Click Access Control, *realm-name*, Subjects, and then Group.**

b. **Click New.**

c. **In ID, enter the name of the group. For example: `manager` or `admin`**

d. **Click OK.**

Repeat these steps for the other group.

3 **To test access to the manager and host-manager applications, add several test users to each group.**

4 **Create a policy with two rules and allow access to the `manager` and `admin` groups. For example:**

```
http://sso-host.example.com:8080/host-manager/*
```

```
http://sso-host.example.com:8080/manager/*
```

5 **If the redirect loop issue is a concern, set the Cookie Encode property to Yes in the OpenSSO Enterprise server:**

a. **In the console, click Configuration, Server and Sites, and the OpenSSO Enterprise Server Instance name.**

b. **Click Security and then Cookie. By default Encode Cookie is set to No.**

c. **Click Inheritance Settings, deselect Encode Cookie, and then click Save.**

You can now change the cookie encoding option.

d. **Click Back to Server Profile.**

   e. **Set Cookie Encoding to Yes and click Save.**

6 **Make the following Tomcat 6.0 manager and host-manager application changes:**

   a. **For the manager application, in the** `$CATALINA_HOME/webapps/manager/WEB-INF/web.xml`**, change** `<role-name>manager</role-name>` **to:**

   ```
   <role-name>id=manager,ou=group,dc=opensso,dc=java,dc=net</role-name>
   ```

   b. **For the host-manager application, in the** `$CATALINA_HOME/webapps/host-manager/WEB-INF/web.xml`**, change** `<role-name>admin</role-name>` **to:**

   ```
   <role-name>id=admin,ou=group,dc=opensso,dc=java,dc=net</role-name>
   ```

   Note. The `dc=opensso,dc=java,dc=net` part in the manager and admin role values is used because OpenSSO Enterprise is deployed using the default mode. If you have a custom setup and the `DN` is different, change the value for your deployment.

7 **Restart the OpenSSO Enterprise server.**

Deploying the Agent Application

The agent application (`agentapp.war`) is a housekeeping application used by the agent for notifications and other functions such as cross domain single sign-on (CDSSO) support.

▼ To Deploy the Agent Application

1 **The agent application (**`agentapp.war`**) is bundled with the** `tomcat_v6_agent_3.zip` **distribution file and is available as follows after you unzip the file:**

PolicyAgent-base`/etc/agentapp.war`

2 **Deploy** `agentapp.war` **on the Tomcat 6.0 instance by copying** `agentapp.war` **to the Tomcat 6.0** `webapps` **directory.**

Important: You must use the same deployment URI that you specified for the "Agent URL" prompt during the agent installation. For example, if you accepted the default value (`/agentapp`) as the deployment URI for the agent application, use this same URI to deploy `agentapp.war`.

Configuring Tomcat Applications Protected by the Tomcat 6.0 Version 3.0 Agent

- "Installing the Agent Filter for a Deployed Application Protected by the Tomcat 6.0 Version 3.0 Agent" on page 22

Installing the Agent Filter for a Deployed Application Protected by the Tomcat 6.0 Version 3.0 Agent

This task is required depending on how you answered the `Install policy agent in global web.xml file` prompt during the Tomcat 6.0 Version 3.0 agent installation:

- `false`: This task is required. Install the agent filter by modifying the deployment descriptor of each application that you want to protect.
- `true`: The task is not required.

▼ To Install the Agent Filter for a Deployed Application Protected by the Tomcat 6.0 Version 3.0 Agent

1 **Ensure that the application you want to protect is not currently deployed on Tomcat 6.0.**

If the application is deployed, undeploy it before continuing.

2 **Backup the application's `web.xml` file before you modify the descriptors.**

The backup copy can be useful if you need to uninstall the agent.

3 **Edit the application's descriptors in the `web.xml` file:**

a. **Set the `<DOCTYPE>` element as shown in the following example:**

```
<!DOCTYPE web-app version="2.4"
xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
```

Note: Tomcat 6.0 supports the Java Servlet specification version 2.4. Version 2.4 is fully backward compatible with version 2.3. Therefore, all existing servlets should work without modification or recompilation.

b. **Add the `<filter>` elements to the deployment descriptor.**

Specify the agent filter as the first `<filter>` element and the agent filter mapping as the first `<filter-mapping>` element. For example:

```
<web-app>
...
    <filter>
        <filter-name>Agent</filter-name>
        <filter-class>com.sun.identity.agents.filter.AmAgentFilter</filter-class>
    </filter>

    <filter-mapping>
        <filter-name>Agent</filter-name>
```

```
            <url-pattern>/*</url-pattern>
            <dispatcher>REQUEST</dispatcher>
            <dispatcher>INCLUDE</dispatcher>
            <dispatcher>FORWARD</dispatcher>
            <dispatcher>ERROR</dispatcher>
        </filter-mapping>
    ...
    </web-app>
```

4 **Restart the Tomcat 6.0 web container.**

5 **Deploy (or redeploy) the application on the Tomcat 6.0 web container.**

The agent filter is then added to the application.

Note: You can also protect an application with Java EE declarative security. To learn more about protecting your application with Java EE declarative security, consider "Deploying the Java EE Policy Agent Sample Application" on page 29.

Optional Post-Installation Tasks for the Tomcat 6.0 Version 3.0 Agent

- "Changing the Password for an Agent Profile" on page 23
- "Creating the Necessary URL Policies" on page 24
- "Configuring J2EE Declarative Security for Tomcat 6.0 Web Applications" on page 25
- "Deploying the Java EE Policy Agent Sample Application" on page 29
- "Configuring Web Services Security for the Tomcat 6.0 Version 3.0 Agent" on page 30

Changing the Password for an Agent Profile

After you install the agent, you can change the agent profile password, if required for your deployment.

▼ To Change the Password for an Agent Profile

1 **On the OpenSSO Enterprise server:**

a. **Login into the OpenSSO Administration Console.**

b. **Click Access Control, *realm-name*, Agents, J2EE, and then the name of the agent profile you want to update.**

The Console displays the Edit page for the agent profile.

c. **Enter and confirm the new unencrypted password.**

d. **Click Save.**

2 **On the server where the Tomcat 6.0 version 3.0 agent is installed:**

a. **In the agent profile password file, replace the old password with the new unencrypted password.**

b. **Change to the** *PolicyAgent-base*/bin **directory.**

c. **Encrypt the new password using the** agentadmin --encrypt **command following this syntax.**

agentadmin --encrypt *agent-instance password-file*

For example:

```
# ./agentadmin --encrypt Agent_001 tomcat6agentpw
```

The agentadmin --encrypt command returns the new encrypted password. For example:

```
ASEWEJIowNBJHTv1UGD324kmT==
```

d. **In the** *agent-instance*/config/OpenSSOAgentBootstrap.properties **file, set the following property to the new encrypted password from the previous step. For example:**

```
com.iplanet.am.service.secret=ASEWEJIowNBJHTv1UGD324kmT==
```

e. **Restart the Tomcat 6.0 instance that is being protected by the policy agent.**

Creating the Necessary URL Policies

If the Tomcat 6.0 version 3.0 agent is configured to operate in the URL_POLICY or ALL filter mode, you must create the appropriate URL policies. For instance, if the agent is available on port 8080 using the HTTP protocol, you must create at minimum, a policy to allow access to the following resource:

```
http://myhost.mydomain.com:8080/agentsample
```

where agentsample is the context URI for the sample application.

If no policies are defined and the agent is configured to operate in the URL_POLICY or ALL filter mode, then no user is allowed access to the resources protected by the Tomcat 6.0 version 3.0 agent.

For more information, see:

- Agent sample application readme.txt file in the /sampleapp directory

- *Sun OpenSSO Enterprise 8.0 Administration Guide* to create these policies using the OpenSSO Enterprise Console or command-line utilities

Configuring J2EE Declarative Security for Tomcat 6.0 Web Applications

This section describes how to configure J2EE declarative security for the Tomcat 6.0 Manager, Administration, and Host Manager web applications, including:

- "Setting the Agent Filter Modes" on page 25
- "Creating OpenSSO Enterprise Users and Groups" on page 26
- "Allowing an OpenSSO Enterprise User to Access the Manager Web Application" on page 27
- "Allowing an OpenSSO Enterprise User to Access the Administration Web Application" on page 27
- "Allowing an OpenSSO Enterprise User to Access the Host Manager Web Application" on page 28

Setting the Agent Filter Modes

By default, the Tomcat 6.0 version 3.0 agent protects the Tomcat Manager, Administration, and Host Manager web applications with J2EE security. This default configuration is set by the agent installer, which sets the Agent Filter Mode (`com.sun.identity.agents.config.filter.mode` property) to `J2EE_POLICY` in the Tomcat 6.0 version 3.0 agent configuration.

If you prefer, you can protect the Manager, Administration, and Host Manager web applications with a filter mode other than `J2EE_POLICY`, depending on the requirements for your deployment. For example, you can change the filter mode for these applications to `URL_POLICY` or `ALL`.

▼ To Set the Agent Filter Modes

1. **Log in to the OpenSSO Enterprise Administration Console.**

2. **Click Access Control,** *realm-name***, Agents, J2EE, and then the name of the Tomcat 6.0 version 3.0 agent.**

3. **Click General and add the Agent Filter Mode as required by your deployment for:**

   - Manager web application (`manager`)
   - Administration web application (`admin`)
   - Host Manager web application (`host-manager`)

The corresponding properties are:

```
com.sun.identity.agents.config.filter.mode[manager]
com.sun.identity.agents.config.filter.mode[admin]
com.sun.identity.agents.config.filter.mode[host-manager]
```

4 **Click Save.**

5 **The `com.sun.identity.agents.config.filter.mode` property is not hot-swappable, so you must restart the OpenSSO Enterprise web container for the new values to take effect.**

Creating OpenSSO Enterprise Users and Groups

In this task, you create new OpenSSO Enterprise users and groups who will be able to access the Tomcat 6.0 version 3.0 Manager, Administration, and Host Manager web applications.

Note – In Access Manager 7.1 and Access Manager 7 2005Q4, users were assigned specific roles. OpenSSO Enterprise uses groups rather than roles for the same functionality.

▼ To Create OpenSSO Enterprise Users and Groups

1 **Log in to the OpenSSO Enterprise Administration Console.**

2 **Click Access Control,** *realm-name*, **Subjects, and then User.**

3 **Create the following new users, as required by your deployment:**

- Manager user: A user who will be assigned to the `manager` group and will be able to log into the Manager web application.
- Administrator user: A user who will be assigned to the `admin` group and will be able to log into the Administration and Host Manager web applications.

4 **Click Access Control,** *realm-name*, **Subjects, and then Group.**

5 **Create the following new groups:**

- Manager group named `manager`
- Administrator group named `admin`

6 **Assign the new users created in Step 3 to their respective groups:**

- Assign the Manager user to the Manager group (`manager`).
- Assign the Administrator user to the Administrator group (`admin`).

7 **Save all changes to the new users and groups.**

Allowing an OpenSSO Enterprise User to Access the Manager Web Application

In this task, you edit the Tomcat 6.0 version 3.0 Manager `web.xml` file to allow an OpenSSO Enterprise user to access the Manager web application.

▼ To Allow an OpenSSO Enterprise User to Access the Manager Web Application

1. **Change to the following directory for the Tomcat 6.0 version 3.0 instance:**

   `$CATALINA_HOME/server/webapps/manager/WEB-INF`

2. **In the `web.xml` file, find the user and role information for the Manager role.**

   This role is defined in the `<role-name>` element under the `<security-role>` element.

3. **Delete the Manager security role.**

4. **Create a new Manager security role using the user and group that you created in the OpenSSO Enterprise Console, as described in "Creating OpenSSO Enterprise Users and Groups" on page 26.**

   For example:

   ```
   <security-role>
   id=manager,ou=group,dc=realm-name,dc=example,dc=com
   </security-role>
   ```

5. **Replace the Manager role defined in the `<role-name>` element under the `<auth-constraint>` element with the contents of the `<role-name>` element as described in the previous step.**

   For example:

   ```
   <auth-constraint>
   <role-name>id=manager,ou=group,dc=realm-name,dc=example,dc=com</role-name>
   </auth-constraint>
   ```

6. **Save the `web.xml` file.**

Allowing an OpenSSO Enterprise User to Access the Administration Web Application

In this task, you edit the Administration web application's `web.xml` file to allow an OpenSSO Enterprise user to access the Administration web application.

▼ To Allow an OpenSSO Enterprise User to Access the Administration Web Application

1. **Change to the following directory for the Tomcat 6.0 version 3.0 instance:**

   `$CATALINA_HOME/server/webapps/admin/WEB-INF`

2. **In the `web.xml` file, find the user and role information for the Administrator role.**

   This role is defined in the `<role-name>` element under the `<security-role>` element.

3. **Delete the Administrator security role.**

4. **Create a new Administrator security role using the user and group that you created in the OpenSSO Enterprise Console, as described in "Creating OpenSSO Enterprise Users and Groups" on page 26.**

   For example:

   ```
   <security-role>
   <role-name>id=admin,ou=group,dc=realm-name,dc=example,dc=com</role-name>
   </security-role>
   ```

5. **Replace the Administrator role defined in the `<role-name>` element under the `<auth-constraint>` element with the contents of the `<role-name>` element as described in the previous step.**

   For example:

   ```
   <auth-constraint>
   <role-name>id=admin,ou=group,dc=realm-name,dc=example,dc=com</role-name>
   </auth-constraint>
   ```

6. **Save the `web.xml` file.**

7. **Restart the Tomcat 6.0 web container.**

Allowing an OpenSSO Enterprise User to Access the Host Manager Web Application

In this task, you edit the Host Manager web application's `web.xml` file to allow an OpenSSO Enterprise user to access the Host Manager web application.

The steps to configuring the Host Manager web application with declarative security are similar to the steps for the Administration web application. Both applications are accessible by users assigned to the `admin` group.

▼ To Allow an OpenSSO Enterprise User to Access the Host Manager Web Application

1. **Change to the following directory for the Tomcat 6.0 version 3.0 instance:**

   `$CATALINA_HOME/server/webapps/host-manager/WEB-INF`

2. **In the `web.xml` file, find the user and role information for the Host Manager web application role.**

   This role is defined in the `<role-name>` element under the `<security-role>` element.

3. **Delete the Host Manager web application security role.**

4. **Create a new Host Manager web application security role using the user and group that you created in the OpenSSO Enterprise Console, as described in "Creating OpenSSO Enterprise Users and Groups" on page 26.**

   For example:

   ```
   <security-role>
   <role-name>id=host-manager,ou=group,dc=realm-name,dc=example,dc=com</role-name>
   </security-role>
   ```

5. **Replace the Host Manager web application role defined in the `<role-name>` element under the `<auth-constraint>` element with the contents of the `<role-name>` element as described in the previous step.**

   For example:

   ```
   <auth-constraint>
   <role-name>id=host-manager,ou=group,dc=realm-name,dc=example,dc=com</role-name>
   </auth-constraint>
   ```

6. **Save the `web.xml` file.**

7. **Restart the Tomcat 6.0 web container.**

Deploying the Java EE Policy Agent Sample Application

Deploying the policy agent sample application is optional. However. after you install the Tomcat 6.0 version 3.0 agent, consider deploying the sample application to help you better understand the key features, functions, and configuration options of Java EE agents, including:

- Single sign-on (SSO)
- Web-tier declarative security
- Programmatic security

- URL policy evaluation
- Session, policy, and profile attribute fetch

The sample application can be especially useful if you are writing a custom agent application.

After you install the Tomcat 6.0 version 3.0 agent, the sample application is available as:

PolicyAgent-base/`sampleapp/dist/agentsample.war`

For information about compiling, deploying, and running the sample application, see the `readme.txt` file in the `/sampleapp` directory.

Configuring Web Services Security for the Tomcat 6.0 Version 3.0 Agent

The Tomcat 6.0 version 3.0 agent supports Web Services Security for web service providers. A web service provider (WSP) deployed on Tomcat 6.0 protected by the agent can have additional security provided by the agent. For example, you can configure the Tomcat 6.0 version 3.0 agent and OpenSSO Enterprise server to support various Web Services Security profiles, including Username token, X509 token, and SAML2 token.

▼ To Configure Web Services Security for the Tomcat 6.0 Agent

You must first download and install the JAX-WS JAR files from the JAX-WS Reference Implementation (RI) project.

Note About the Examples. The examples in this section use `/opt` as the download and installation directory. However, if you prefer, you can use a different directory. These examples are also intended for a Solaris or Linux system. If you are running on another platform such as Windows, you will need to make changes for the paths and filenames.

1. **Download and unzip Tomcat 6.0 in `/opt`. For example:** `/opt/apache-tomcat-6.0.18`

2. **Download `jaxws-ri.zip` from the following site:** `https://jax-ws.dev.java.net/`

3. **Unzip `jaxws-ri.zip`, also in `/opt`.**

4. **On Solaris and Linux systems, set the JAX-WS RI shell scripts to be executable. For example:**

   ```
   cd /opt/jaxws-ri/bin
   chmod +x *.sh
   ```

5. **In `/opt/jaxws-ri/tomcat.xml`, modify the `tomcat.home` property for your deployment. For example:**

   ```
   <property name="tomcat.home" value="/opt/apache-tomcat-6.0.18"/>
   ```

6 **Install the JAX-WS JAR files into Tomcat 6.0. For example, using `ant`:**

```
/share/builds/components/ant/1.6.5/bin/ant -f tomcat.xml install
```

7 **Configure and deploy your WSP application.**

If you are deploying new web services that uses JAX-WS, see "Configuring the `StockService` and `StandAloneStockClient` Samples" on page 31 as an example to follow for your web services.

If your application is already deployed and using WSS with JAX-WS, you might need only to add the agent filter in the `web.xml` file.

8 **Install and configure the Tomcat 6.0 version 3.0 agent, as described in this guide.**

9 **Follow the general steps to configure the web service provider (WSP) and web service client (WSC) in "Web Services Security Support for J2EE Agents in Policy Agent 3.0" in *Sun OpenSSO Enterprise Policy Agent 3.0 User's Guide for J2EE Agents*.**

10 **Deploy your WSP application into the Tomcat 6.0 web container. For example, to deploy the `StockService.war` sample:**

```
cp /opt/wsp/samples/tomcat/StockService/dist/StockService.war
/opt/apachetomcat-6.0.18/webapps/
```

11 **Start the Tomcat 6.0 web container.**

Configuring the `StockService` and `StandAloneStockClient` Samples

This section describes how to configure the Sun `StockService` sample as the WSP and the `StandAloneStockClient` as the WSC. Use these samples as models to configure your own WSS applications.

- "To Configure the `StockService` Sample" on page 31
- "To Configure the `StandAloneStockClient` Sample" on page 35

▼ To Configure the `StockService` Sample

1 **Create the `wsp` directory. For example, in `/opt`.**

2 **Download `openssowssproviders.zip` from the `WSS Agent` link on `https://opensso.dev.java.net/public/use/index.html`.**

3 **Unzip `openssowssproviders.zip` in `/opt/wsp/`.**

4 **Create the `tomcat` directory under `/opt/wsp/samples` for the Tomcat 6.0 files. For example:**

```
cd /opt/wsp/samples
mkdir tomcat
```

5. **Copy the GlassFish sample files to the new `tomcat` directory:**

   ```
   cp -r /opt/wsp/samples/glassfish/* /opt/wsp/samples/tomcat/
   ```

6. **Rename `glassfish.properties` for Tomcat 6.0:**

   ```
   cd /opt/wsp/samples/tomcat/
   mv glassfish.properties tomcat.properties
   ```

7. **In `/opt/wsp/samples/tomcat/tomcat.properties`, remove the GlassFish properties and add the following:**

   ```
   wsp.home=/opt/wsp
   jaxws.home=/opt/jaxws-ri
   jaxws.lib.dir=/opt/jaxws-ri/lib
   ```

8. **Edit `/opt/wsp/samples/tomcat/StockService/build.xml` as shown in the next example.**

Tip – To create a new Tomcat `build.xml` file, just copy the following XML statements.

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="StockQuoteService" default="all" basedir=".">
  <description>Builds, tests, and runs the project stockclient.</description>
  <property file="../tomcat.properties"/>
  <condition property="wsimport-script-suffix" value=".bat">
    <os family="windows"/>
  </condition>
  <condition property="wsimport-script-suffix" value=".sh">
    <not>
      <os family="windows"/>
    </not>
  </condition>
  <path id="build.class.path">
    <pathelement location="build/classes"/>
    <fileset dir="${jaxws.lib.dir}">
      <include name="**/*.jar"/>
    </fileset>
  </path>
  <target name="-pre-compile">
    <mkdir dir="build/classes"/>
    <mkdir dir="web/WEB-INF/classes"/>
    <exec executable="${jaxws.home}/bin/wsimport${wsimport-script-suffix}">
    <arg line="-verbose -d build/classes web/WEB-INF/wsdl/StockService/stockservice.wsdl"/>
    </exec>
    <copy file="src/java/handlers.xml" todir="web/WEB-INF/classes"/>
  </target>
    <target name="compile" depends="-pre-compile">
    <javac fork="true" destdir="build/classes" srcdir="src/java">
```

```
    <classpath refid="build.class.path" />
    </javac>
  </target>
  <target name ="war" depends="compile">
    <mkdir dir="dist"/>
    <copy todir="web/WEB-INF/classes">
    <fileset dir="build/classes" />
    </copy>
    <war destfile="dist/StockService.war" webxml="web/WEB-INF/web.xml">
    <zipfileset dir="web" />
    </war>
  </target>
    <target name="all">
    <antcall target="war" />
  </target>
</project>
```

9 **In the following file, change the references to `localhost` and port `8080`, depending on your deployment:**

```
/opt/wsp/samples/jboss/StockService/web/WEB-INF/wsdl/StockService/stockservice.wsdl
```

10 **Remove `/opt/wsp/samples/jboss/StockService/web/WEB-INF/sun-web.xml`. For example:**

```
cd /opt/wsp/samples/jboss/StockService/web/WEB-INF
rm sun-web.xml
```

11 **In the same directory, create `sun-jaxws.xml` with the following content:**

```
<?xml version="1.0" encoding="UTF-8"?>
<endpoints
  xmlns='http://java.sun.com/xml/ns/jax-ws/ri/runtime'
  version='2.0'>
  <endpoint
    name='StockService'
    implementation='com.samples.StockService'
    url-pattern='/StockService' />
</endpoints>
```

12 **In the same directory, in `web.xml`, add the agent `<filter>`, `<filter-mapping>`, `<listener>`, `<servlet>`, and `<servlet-mapping>` entries, as follows:**

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
    version="2.5">

  <filter>
```

```
    <filter-name>Agent</filter-name>
    <filter-class> com.sun.identity.agents.filter.AmAgentFilter </filter-class>
  </filter>
  <filter-mapping>
    <filter-name>Agent</filter-name>
    <url-pattern>/*</url-pattern>
    <dispatcher>REQUEST</dispatcher>
   <dispatcher>INCLUDE</dispatcher>
    <dispatcher>FORWARD</dispatcher>
    <dispatcher>ERROR</dispatcher>
  </filter-mapping>

  <session-config>
    <session-timeout>
      30
    </session-timeout>
  </session-config>
  <welcome-file-list>
    <welcome-file>
      index.jsp
    </welcome-file>
  </welcome-file-list>

  <listener>
    <listener-class>
      com.sun.xml.ws.transport.http.servlet.WSServletContextListener
    </listener-class>
  </listener>

  <servlet>
    <description>JAX-WS endpoint</description>
    <display-name>The JAX-WS servlet</display-name>
    <servlet-name>jaxws</servlet-name>
    <servlet-class>com.sun.xml.ws.transport.http.servlet.WSServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>jaxws</servlet-name>
    <url-pattern>/StockService</url-pattern>
  </servlet-mapping>
</web-app>
```

13 **Build the `StockService` WAR file. For example, using `ant`:**

```
cd /opt/wsp/samples/jboss/StockService
/share/builds/components/ant/1.6.5/bin/ant -f build.xml
```

▼ To Configure the `StandAloneStockClient` Sample

1. **Change to the `StandAloneStockClient` directory:**

   ```
   cd /opt/wsp/samples/tomcat/StandAloneStockClient
   ```

2. **In `src/com/samples/SecuringWS.java`, change any references to `localhost` and port `8080`, depending on your deployment.**

3. **In the `/opt/wss/samples/tomcat/StandAloneStockClient` directory, modify `build.xml` for Tomcat 6.0 rather than GlassFish:**

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="StandAloneStockClient" default="default" basedir=".">
  <description>Builds, tests, and runs the project stockclient.</description>
  <property file="../tomcat.properties"/>
  <property name="is.java-client.module" value="true"/>
  <target name="default" depends="run"/>
  <target name="build" depends="clean">
    <mkdir dir="build/classes"/>
    <javac srcdir="src"
    destdir="build/classes"
    classpath="xyz.jar"
    debug="on">
    <classpath>
      <pathelement location="${wsp.home}/lib/openssowssproviders.jar"/>
      <pathelement location="${wsp.home}/lib/webservices-rt.jar"/>
      <pathelement location="${wsp.home}/lib/openssoclientsdk.jar"/>
      <pathelement location="${wsp.home}/lib/xalan.jar"/>
      <pathelement location="${wsp.home}/lib/xercesImpl.jar"/>
      <pathelement location="${wsp.home}/lib/j2ee.jar"/>
      <pathelement location="${wsp.home}/lib"/>
      <pathelement path="build/classes"/>
    </classpath>
    </javac>
</target>

<target name="run" depends="build">
    <echo>java.home=${java.home}</echo>
  <java classname="com.samples.SecuringWS" fork="true">
    <classpath>
      <pathelement location="${wsp.home}/lib/openssowssproviders.jar"/>
      <pathelement location="${wsp.home}/lib/ldapjdk.jar"/>
      <pathelement location="${wsp.home}/lib/webservices-rt.jar"/>
      <pathelement location="${wsp.home}/lib/openssoclientsdk.jar"/>
      <pathelement location="${wsp.home}/lib/xalan.jar"/>
      <pathelement location="${wsp.home}/lib/xercesImpl.jar"/>
      <pathelement location="${wsp.home}/lib/j2ee.jar"/>
      <pathelement location="${wsp.home}/lib"/>
```

```
      <pathelement path="build/classes"/>
    </classpath>
    </java>
  </target>

  <target name="clean">
    <delete dir="dist"/>
    <delete dir="build"/>
  </target>
</project>
```

4 **Modify** `/opt/wsp/lib/AMConfig.properties` **depending on your setup, so that the** `StandAloneStockClient` **sample sends a secure web service request:**

```
com.iplanet.services.debug.level=error
com.iplanet.services.debug.directory=/tmp/wss
com.iplanet.am.naming.url=http://opensso-host:port/opensso/namingservice
com.sun.identity.agents.app.username=amadmin
com.iplanet.am.service.password=amadmin-password
com.iplanet.am.service.secret=
am.encryption.pwd=
com.sun.identity.client.encryptionKey=
com.iplanet.am.server.protocol=http
com.iplanet.am.server.host=opensso-host
com.iplanet.am.server.port=port
com.iplanet.am.services.deploymentDescriptor=/opensso
com.iplanet.am.cookie.name=iPlanetDirectoryPro
com.sun.identity.saml.xmlsig.keystore=/opt/wsp/resources/keystore.jks
com.sun.identity.saml.xmlsig.storepass=/opt/wsp/resources/.storepass
com.sun.identity.saml.xmlsig.keypass=/opt/wsp/resources/.keypass
com.sun.identity.saml.xmlsig.certalias=cert-alias
com.sun.identity.loginurl=http://your-opensso-hostname:port/opensso/UI/Login
com.sun.identity.liberty.authnsvc.url=http://opensso-host:port/opensso/Liberty/authnsvc
```

5 **Execute the** `StandAloneStockClient`**. For example:**

```
/share/builds/components/ant/1.6.5/bin/ant -f build.xml.
```

You should see the requests and responses. Also, check the Tomcat 6.0 agent debug file.

Managing the Tomcat 6.0 Version 3.0 Agent

OpenSSO Enterprise stores version 3.0 policy agent configuration data (as well as server configuration data) in a centralized repository. To manage this configuration data, use these options:

- OpenSSO Enterprise Administration Console

You can manage both version 3.0 Java EE and web agents from the OpenSSO Enterprise Console. Tasks that you can perform include creating, deleting, updating, listing, and displaying agent configurations. Using the Console, you can set properties for an agent that you previously set by editing the agent's `AMAgent.properties` file.

For more information, refer to the Administration Console online Help.

- `ssoadm` command-line utility

  The `ssoadm` utility is the command-line interface to OpenSSO Enterprise server and is available after you install the tools and utilities in the `ssoAdminTools.zip` file. The `ssoadm` utility includes subcommands to manage policy agents, including:

  - Creating, deleting, updating, listing, and displaying agent configurations
  - Creating deleting, listing, and displaying agent groups
  - Adding and removing an agent to and from a group

  For information about the `ssoadm` utility, including the syntax for each subcommand, see the *Sun OpenSSO Enterprise 8.0 Administration Reference*.

Managing a Version 3.0 Agent With a Local Configuration

In some scenarios, you might need to deploy a version 3.0 agent using a local configuration. For example, if you deploy the agent with Access Manager 7.1 or Access Manager 7 2005Q4, which do not support centralized agent configuration, local configuration is used by default.

In this scenario, you must manage the version 3.0 agent by editing properties in the agent's local `OpenSSOAgentConfiguration.properties` file (in the same manner that you edit the `AMAgent.properties` file for version 2.2 agents).

Caution – A version 3.0 agent also stores configuration information in the local `OpenSSOAgentBootstrap.properties` file. The agent uses information in the bootstrap file to start and initialize itself and to communicate with OpenSSO Enterprise server. In most cases, you won't need to edit the bootstrap file; however, if you do edit the file, be very careful, or the agent might not function properly.

Uninstalling the Tomcat 6.0 Version 3.0 Agent

- "Preparing to Uninstall the Tomcat 6.0 Version 3.0 Agent" on page 38
- "Uninstalling the Tomcat 6.0 Version 3.0 Agent Using the `agentadmin` Program" on page 38

Preparing to Uninstall the Tomcat 6.0 Version 3.0 Agent

▼ To Prepare to Uninstall Tomcat 6.0 Version 3.0 Agent

1. **Undeploy any applications protected by the Tomcat 6.0 version 3.0 agent.**

2. **Restore the deployment descriptors of these applications to their original deployment descriptors. (Backup files are useful here if you have them.)**

3. **Conditionally, if you are permanently removing the Tomcat 6.0 version 3.0 agent, undeploy the agent application.**

   However, if you plan to re-install this agent , you don't need to undeploy the agent application.

4. **Ensure that the Tomcat 6.0 instance that is being protected by the agent is stopped.**

Uninstalling the Tomcat 6.0 Version 3.0 Agent Using the `agentadmin` Program

▼ To Uninstall the Tomcat 6.0 Version 3.0 Agent

1. **Change to the following directory:**

   PolicyAgent-base`/bin`

2. **Issue one of the following commands:**

   ```
   # ./agentadmin --uninstall
   ```

   or

   ```
   # ./agentadmin --uninstallAll
   ```

   The `--uninstall` option removes only one instance of the agent, while the `--uninstallAll` option prompts you to remove all configured instances of the agent.

3. **The `uninstall` program prompts you for the configuration directory path of the Tomcat 6.0 instance.**

4. **The `uninstall` program displays a summary of your responses and then asks if you want to continue:**

   To continue with the uninstallation, select 1 (the default).

Example 2 Uninstallation Sample for the Tomcat 6.0 Version 3.0 Agent

```
*************************************************************************
Welcome to the OpenSSO Policy Agent for Apache Tomcat 6.0 Servlet/JSP
Container
*************************************************************************
Enter the complete path to the directory which is used by Tomcat Server to
store its configuration Files. This directory uniquely identifies the
Tomcat Server instance that is secured by this Agent.
[ ? : Help, ! : Exit ]
Enter the Tomcat Server Config Directory Path
[/opt/apache-tomcat-6.0.18/conf]:
-----------------------------------------------
SUMMARY OF YOUR RESPONSES
-----------------------------------------------
Tomcat Server Config Directory : /opt/apache-tomcat-6.0.18/conf
Verify your settings above and decide from the choices below.
1. Continue with Uninstallation
2. Back to the last interaction
3. Start Over
4. Exit
Please make your selection [1]:
Removing the agent classpath from
/opt/apache-tomcat-6.0.18/bin/setclasspath.sh script ...DONE.
Deleting the config directory
/opt/agents/j2ee_agents/tomcat_v6_agent/Agent_001/config ...DONE.
Removing OpenSSO Tomcat Agent Realm from Server XML file :
/opt/apache-tomcat-6.0.18/conf/server.xml ...DONE.
Removing filter from Global deployment descriptor file :
/opt/apache-tomcat-6.0.18/conf/web.xml ...DONE.
Removing OpenSSO Tomcat Agent Filter and Form login authentication from Web
applications ...DONE.

Uninstall log file location:
/opt/agents/j2ee_agents/tomcat_v6_agent/installer-logs/audit/uninstall.log
Thank you for using OpenSSO Policy Agent
```

After You Finish the Uninstall

- The `/config` directory is removed from the agent instance directory, but the `/installer-logs` directory still exists.
- The `uninstall` program creates an uninstall log file in the *PolicyAgent-base*`/installer-logs/audit` directory.
- The agent instance directory is not automatically removed. For example, if you uninstall the agent for Agent_001, a subsequent agent installation creates the Agent_002 instance directory. To remove an agent instance directory, you must manually remove the directory.

Migrating a Version 2.2 Policy Agent

The version 3.0 `agentadmin` program includes the new `--migrate` option to migrate a version 2.2 agent to version 3.0. After you migrate a version 2.2 agent, the agent can use the new version 3.0 agent features.

The migration process migrates the agent's binary files, updates the agent's deployment container configuration, and converts the agent's `AMAgent.properties` file to the new version 3.0 `OpenSSOAgentBootstrap.properties` and `OpenSSOAgentConfiguration.properties` files.

Migrating a version 2.2 agent involves these general steps:

1. On the server where the version 2.2 agent is installed, run the version 3.0 `agentadmin` program with the `--migrate` option.

   To get the version 3.0 `agentadmin` program, you must download the version 3.0 agent that corresponds to the version 2.2 agent you are migrating. For example, if you are migrating the version 2.2 Apache Tomcat agent, download the version 3.0 Tomcat 6.0 version 3.0 agent.
2. On the OpenSSO Enterprise server, run the `ssoadm` utility to create the new version 3.0 agent configuration in the centralized agent configuration repository.

   Therefore, the `ssoadm` utility must be installed from the `ssoAdminTools.zip` file on the OpenSSO Enterprise server. For information, see "Installing the OpenSSO Enterprise Utilities and Scripts" in the *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

The `agentadmin` program creates a new deployment directory for the migrated agent, starting with Agent_001. The program does not modify the version 2.2 agent deployment directory files, in case you need these files after you migrate.

The following procedure, the migrated version 3.0 agent instance uses a new agent profile name, which is Tomcat6v3Agent in the examples. The old version 2.2 and new version 3.0 agent profile passwords are the same. If you need to change the password for the new version 3.0 agent profile, see "Changing the Password for an Agent Profile" on page 23.

▼ To Migrate a Version 2.2 Agent:

1 **Login to the server where the version 2.2 agent is installed.**

To migrate the agent, you must have write permission to the version 2.2 agent's deployment container files and directories.

2 **Stop the Tomcat instance for the version 2.2 agent.**

3 **Create a directory to download and unzip the version 3.0 agent. For example:** `v30agent`

4 **Download and unzip the version 3.0 agent that corresponds to the version 2.2 agent you are migrating.**

The version 3.0 agents are available from the OpenSSO project site: `https://opensso.dev.java.net/public/use/index.html`

5 **Change to the version 3.0 agent's `/bin` directory.**

For example, if you downloaded and unzipped the version 3.0 Tomcat 6.0 version 3.0 agent in the `v30agent` directory:

```
cd /v30agent/j2ee_agents/tomcat_v6_agent/bin
```

6 **On Solaris and Linux systems, set the permissions for the `agentadmin` program as follows, if needed:**

```
# chmod 755 agentadmin
```

7 **Run the version 3.0 `agentadmin` program with the `--migrate` option. For example:**

```
./agentadmin --migrate
```

8 **When the `agentadmin` program prompts you, enter the path to the version 2.2 agent's deployment directory. For example:**

```
...
Enter the migrated agent’s deployment directory:
/opt/j2ee_agents/tomcat_v6_agent
...
```

In this example, `/opt` is the directory where you downloaded and upzipped the version 2.2 agent.

The `agentadmin` program migrates the version 2.2 agent.

9 **After the `agentadmin` program finishes, set the following properties:**

a. **In `Agent_`*nnn*`/config/OpenSSOAgentBootstrap.properties`, change:**

`com.sun.identity.agents.config.username =` *new-v3.0-agent-profile-name*

For example:

```
com.sun.identity.agents.config.username = Tomcat6v3Agent
```

10 **Copy the `Agent_`*nnn*`/config/OpenSSOAgentConfiguration.properties` file to the `/bin` directory where `ssoadm` is installed on the OpenSSO Enterprise server.**

11 **In `OpenSSOAgentConfiguration.properties`, add the un-encrypted version 2.2 agent profile password at the end of the file, as follows:**

`userpassword=`*v2.2–agent-profile-password*

12 **On OpenSSO Enterprise server, create a password file for the OpenSSO Enterprise administrator (`amadmin`).**

This password file is an ASCII text file with only one line specifying the `amadmin` password in plain text. For example: `amadminpw`

13 **On OpenSSO Enterprise server, run `ssoadm` to create a new agent configuration in the OpenSSO Enterprise centralized agent configuration repository. For example:**

```
cd tools_zip_root/opensso/bin
./ssoadm create-agent -e / -b Tomcat6v3Agent -t J2EEAgent -u amadmin
-f amadminpw -D ./OpenSSOAgentConfiguration.properties
```

In this example:

- *tools_zip_root* is the directory where you unzipped `ssoAdminTools.zip`.
- `-e /` specifies the specifies the root realm for the agent configuration.
- `-b Tomcat6v3Agent` specifies the version 3.0 agent configuration name.
- `-t J2EEAgent` specifies the agent type for Java EE agents.
- `-u amadmin` species the OpenSSO Enterprise administrator
- `-f amadminpw` specifies the path to the administrator password file.
- `-D ./OpenSSOAgentConfiguration.properties` specifies the agent configuration file

Caution: After you run `ssoadm`, you might want to delete `OpenSSOAgentConfiguration.properties` from the `/bin` directory. This file contains sensitive information, including as the agent profile password, and the original file is maintained on the server where the agent is installed.

14 **Restart the Tomcat instance for the migrated agent.**

Next Steps After you migrate the agent, you can manage the new 3.0 agent configuration using the OpenSSO Enterprise Administration Console or the `ssoadm` utility, as described in "Managing the Tomcat 6.0 Version 3.0 Agent" on page 36.

Sun Related Information

- "Additional Sun Resources" on page 43
- "Accessibility Features for People With Disabilities" on page 43
- "Related Third-Party Web Sites" on page 43
- "How to Report Problems and Provide Feedback" on page 43
- "Sun Welcomes Your Comments" on page 44

Additional Sun Resources

You can find additional useful information and resources at the following locations:

- Sun IT Services: http://www.sun.com/service/consulting/
- Sun Software Products: http://wwws.sun.com/software/
- Sun Support Resources: http://sunsolve.sun.com/
- Sun Developer Network (SDN): http://developers.sun.com/
- Sun Developer Services: http://www.sun.com/developers/support/

Accessibility Features for People With Disabilities

To obtain accessibility features that have been released since the publishing of this media, consult Section 508 product assessments available from Sun upon request to determine which versions are best suited for deploying accessible solutions.

For information about Sun's commitment to accessibility, see http://www.sun.com/accessibility/.

Related Third-Party Web Sites

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

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

How to Report Problems and Provide Feedback

If you have questions or issues with OpenSSO Enterprise, contact Sun as follows:

- Sun Support Resources (SunSolve) services at http://sunsolve.sun.com/.

  This site has links to the Knowledge Base, Online Support Center, and ProductTracker, as well as to maintenance programs and support contact numbers.
- The telephone dispatch number associated with your maintenance contract

So that we can best assist you in resolving problems, please have the following information available when you contact Sun:

- Description of the problem, including when the problem occurs and its impact on your operation
- Machine type, operating system version, web container and version, JDK version, and OpenSSO Enterprise version, including any patches or other software that might be affecting the problem
- Steps to reproduce the problem
- Any error logs or core dumps

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to `http://docs.sun.com/` and click `Feedback`. In the online form, provide the full document title and part number. The part number is a 7-digit or 9-digit number that can be found on the title page or in the document's URL. For example, the title of this guide is *Sun OpenSSO Enterprise Policy Agent 3.0 Guide for Apache Tomcat 6.0*, and the part number is 820-7251.

Revision History

Part Number	Date	Description
820–7251–20	May 14, 2009	Added more configuration information for the agent profile, to allow the manager and host-manager applications to function properly with the agent.
820–7251–10	April 22, 2009	Initial release.