Sun OpenSSO Enterprise Policy Agent 3.0 Guide for IBM WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1

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

Part No: 820–7250–11
July 1, 2009

090702@22510

Sun OpenSSO Enterprise Policy Agent 3.0 Guide for IBM WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1

Last updated July 1, 2009

The WebSphere Application Server/Portal Server policy agent is a version 3.0 Java EE agent (formerly called a J2EE agent) that functions with Sun™ OpenSSO Enterprise to protect resources on IBM® WebSphere® Application Server 6.1, WebSphere Application Server 7.0, and WebSphere Portal Server 6.1.

Contents

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*.

Note – Sun also provides version 2.2 policy agents for WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1. However, to use the new version 3.0 policy agent features, you must deploy the version 3.0 WebSphere Application Server/Portal Server agent described in this guide.

Supported Platforms and Web Containers for the WebSphere Application Server/Portal Server Agent

Supported Versions of WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1

The following versions of WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1 are supported for this agent:

- WebSphere Application Server 7.0

  `http://www-01.ibm.com/software/webservers/appserv/was/`

- WebSphere Application Server 6.1

  `http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp`

- WebSphere Portal Server 6.1

  `http://www-01.ibm.com/software/websphere/portal/`

Note: This guide is written for both WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1. Any differences between the products and versions are noted in the text.

Supported Platforms for the WebSphere Application Server/Portal Server Agent

Agent For	Supported Platforms
■ WebSphere Application Server 7.0 ■ WebSphere Application Server 6.1 ■ WebSphere Portal Server 6.1	■ Solaris OS on SPARC and 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 Server 2003 and 2008, Standard Edition and Enterprise Edition (32-bit and 64-bit platforms) ■ IBM AIX® 5.2, 5.3, and 6.1

- Minor versions of WebSphere Application Server 7.0, WebSphere Application Server 6.1, and WebSphere Portal Server 6.1 are supported.
- Minor versions of the supported platforms, including updates, service packs, and patches, are also supported.

Compatibility and Coexistence for the WebSphere Application Server/Portal Server 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. 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 `http://docs.sun.com/coll/1322.1`.

Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent

Setting Your `JAVA_HOME` Environment Variable

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

Downloading and Unzipping the `websphere_v61_agent_3.zip` Distribution File

▼ To Download and Unzip the `websphere_v61_agent_3.zip` Distribution File

1. **Login to the server where you want to install the agent.**
2. **Create a directory to unzip the `websphere_v61_agent_3.zip` distribution file.**

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

3 **Download and unzip the `websphere_v61_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/websphere_v61_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`). The agent application 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 31.
`/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
`/sampleapp`	Policy agent sample application. For information, see "Deploying the Java EE Policy Agent Sample Application" on page 37.

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 WebSphere Application Server/Portal Server 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/wsas6agentpw`
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.**

Creating an Agent Profile

The WebSphere Application Server/Portal Server agent uses an agent profile to communicate with OpenSSO Enterprise server. You can create an agent profile using any of these three methods:

- Create the agent profile during installation when you run the `agentadmin` program with 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:** `WSASAgentProfile`

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 (`agentapp`).**

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

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. For more information, see "Deploying the Agent Application" on page 31.

8 **Click Create.**

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

To do additional configuration for the agent profile, click the agent link to display the `Edit` agent page. For information about the agent configuration fields, see the Console online Help.

If you prefer, you can also use the `ssoadm` command-line utility to edit the agent profile. For more information, see the *Sun OpenSSO Enterprise 8.0 Administration Reference*.

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 agent using the `agentadmin` program.

---

9 **If the WebSphere Application Server/Portal Server agent will not retrieve the Role from the Access Manager SDK (AMSDK) Identity Repository Plug-in, perform the following steps:**

a. **Click the WebSphere Application Server/Portal Server agent link (for example, `WSASAgentProfile`) to display the agent profile Edit page.**

b. **Click the Application subtab.**

c. **Click the Privilege Attributes Processing link.**

d. **Under the Privilege Attributes Processing section, remove Role from the Current Values for the Privileged Attribute Type list box.**

You should have only Group left under the Current Values.

e. **Click Save.**

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 WebSphere Application Server/Portal Server Agent

Gathering Information to Install the WebSphere Application Server/Portal Server Agent

The following table describes the information you will need to provide when you run the `agentadmin` program to install the WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server Agent

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server 6.1/7.0 instance. Applies to both default and custom installation options. For example: `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/`*hostname*`Node01Cell/nodes/`*hostname*`Node01/servers/server1`
Server Instance name	Name of the WebSphere Application Server 6.1/7.0 instance. Applies only to the custom installation option. Default: `server1`
WebSphere Install Root directory	Installation root directory Applies to both default and custom installation options. Default: `/opt/IBM/WebSphere/AppServer`
URL where OpenSSO Enterprise is running	OpenSSO Enterprise URL, including the deployment URI Applies to both default and custom installation options. For example: `https://openssohost.example.com:8080/opensso`
Agent URL	Agent URL, including the deployment URI Applies to both default and custom installation options. For example: `https://agenthost.example.com:8090/agentapp` The agent application 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 31.
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 8.

TABLE 1 Information Required to Install the WebSphere Application Server/Portal Server Agent *(Continued)*

Prompt	Description
Path to agent profile password file	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.`	If you have already created the agent profile in the OpenSSO Console or by using the `ssoadm` command, enter `false`. 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 10. 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 WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server agent:

- A OpenSSO Enterprise server instance must be installed and running.
- The WebSphere Application Server 6.1/7.0 instance must be installed and configured on the server where you plan to install the agent.

- You must have downloaded and unzipped the distribution file, as described in "Downloading and Unzipping the `websphere_v61_agent_3.zip` Distribution File" on page 6.

▼ To Install the WebSphere Application Server/Portal Server 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 WebSphere Application Server 6.1/7.0 instance files and directories.

2. **If necessary, shut down the WebSphere Application Server 6.1/7.0 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:

   ```
   ---------------------------------------------
   SUMMARY OF YOUR RESPONSES
   ---------------------------------------------
   Instance Config Directory :
   /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
     agenthostNode01Cell/nodes/agenthostNode01/servers/server1
   Instance Server name : server1
   WebSphere Install Root Directory : /opt/IBM/WebSphere/AppServer
   OpenSSO server URL : http://opensso.example.com:8080/opensso
   Agent URL : http://agenthost.example.com:9080/agentapp
   Encryption Key : e/usCNJI2Y57Tyg3S5Wz/5Jc9uxb/ZMn
   ```

```
Agent Profile name : websphere6.1
Agent Profile Password file name : wasagentpw
```

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:

```
SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Bootstrap file location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/
  config/OpenSSOAgentBootstrap.properties
Agent Configuration file location
/agents/j2ee_agents/websphere_v61_agent/Agent_001/
  config/OpenSSOAgentConfiguration.properties
Agent Audit directory location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/logs/audit
Agent Debug directory location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/logs/debug

Install log file location:
/agents/j2ee_agents/websphere_v61_agent/installer-logs/audit/custom.log
Thank you for using OpenSSO Policy Agent
```

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

```
installer-logs/audit
```

9 Restart the WebSphere Application Server 6.1/7.0 instance that is being protected by the agent.

Note – After you install the WebSphere Application Server/Portal Server agent for a specific domain, you cannot use that same agent on the same host for a different domain. To use the WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server Agent

```
***************************************************************************
Welcome to the OpenSSO Policy Agent for IBM WebSphere Application Server 6.1
***************************************************************************

Enter the fully qualified path to the configuration directory of the Server
Instance for the WebSphere node.
```

```
[ ? : Help, ! : Exit ]
Enter the Instance Config Directory
[/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
Node01Cell/nodes/Node01/servers/server1]:
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
  agenthostNode01Cell/nodes/agenthostNode01/servers/server1

Enter the Server Instance name.
[ ? : Help, < : Back, ! : Exit ]
Enter the Server Instance name [server1]:

Enter the WebSphere Install Root directory.
[ ? : Help, < : Back, ! : Exit ]
Enter the WebSphere Install Root directory
[/opt/IBM/WebSphere/AppServer]:

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.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://agenthost.example.com:9080/agentapp

Enter a valid Encryption Key.
[ ? : Help, < : Back, ! : Exit ]
Enter the Encryption Key [e/usCNJI2Y57Tyg3S5Wz/5Jc9uxb/ZMn]:

Enter the Agent profile name
[ ? : Help, < : Back, ! : Exit ]
Enter the Agent Profile name: websphere6.1

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: wasagentpw

---------------------------------------------
SUMMARY OF YOUR RESPONSES
---------------------------------------------
Instance Config Directory :
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
agenthostNode01Cell/nodes/agenthostNode01/servers/server1
Instance Server name : server1
WebSphere Install Root Directory : /opt/IBM/WebSphere/AppServer
```

```
OpenSSO server URL : http://opensso.example.com:8080/opensso
Agent URL : http://agenthost.example.com:9080/agentapp
Encryption Key : e/usCNJI2Y57Tyg3S5Wz/5Jc9uxb/ZMn
Agent Profile name : websphere6.1
Agent Profile Password file name : wasagentpw
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]:
Copy agent.jar,openssoclientsdk.jar to
/opt/IBM/WebSphere/AppServer/lib/ext...DONE.
Creating directory layout and configuring Agent file for Agent_001
instance ...DONE.
Reading data from file wasagentpw 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/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/Node01Cell/nodes/
Node01/servers/server1/server.xml
...DONE.
Configure server.xml file
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/Node01Cell/nodes/
Node01/servers/server1/server.xml...DONE.
Creating the Agent Profile websphere6.1 ...DONE.

SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Bootstrap file location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/config/OpenSSOAgentBootstrap.properties
Agent Configuration file location
/agents/j2ee_agents/websphere_v61_agent/Agent_001/config/OpenSSOAgentConfiguration.properties
Agent Audit directory location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/logs/audit
Agent Debug directory location:
/agents/j2ee_agents/websphere_v61_agent/Agent_001/logs/debug

Install log file location:
/agents/j2ee_agents/websphere_v61_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/websphere_v61_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 WebSphere Application Server/Portal Server Agent

Installing the WebSphere Application Server/Portal Server Agent on Multiple WebSphere Application Server 6.1/7.0 Instances

You can install the WebSphere Application Server/Portal Server agent on multiple WebSphere Application Server 6.1/7.0 instances on the same host machine. However, you must run the agentadmin program for each WebSphere Application Server 6.1/7.0 instance. During each installation, specify the unique server configuration directory and instance name, so the agent can differentiate the different instances.

Installing the WebSphere Application Server/Portal Server Agent on the OpenSSO Enterprise Host Machine

You can install the WebSphere Application Server/Portal Server 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 agent and OpenSSO Enterprise server. For example, on WebSphere Application Server/Portal Server.

Required Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent

Creating an Agent User

The agent user is required for integration with the WebSphere Application Server 6.1/7.0 Console.

▼ To Create an Agent User

1. **Login into the OpenSSO Enterprise Administration Console.**
2. **Click Access Control,** *realm-name*, **Subjects, and then User.**
3. **Click New and enter values for the following fields:**
   - **ID**: Name of the primary agent user. For example: `agentuser`
   - **First Name**: Not required. You can leave this field blank.
   - **Last Name** and **Full Name**: For simplicity, use the same name for each of these values that you specified for ID.
   - **Password** (and confirm): Password for the agent user.

     Note: Be sure to save this password, because you will need to encrypt it later.
   - **User Status**: Active
4. **Click OK.**

   The console creates the agent user and displays the User page again with a link to the new agent user, `agentuser`.

To do additional configuration for the agent profile, click this link to display the Edit User page. For information about the configuration fields, see the Console online Help.

Creating a New Agent Group for the Agent User

The agent profile group is created specifically for the agent user and allows the agent user to access user attributes in the user repository.

Note – In the Access Manager 7.1 and Access Manager 7 2005Q4 releases, the agent user was assigned a specific role. OpenSSO Enterprise uses groups rather than roles for the same functionality.

▼ To Create a New Agent Group for the Agent User

1. **Login to the OpenSSO Enterprise Administration Console.**
2. **Click Access Control and then the name of the realm for which you would like to create the agent group.**
3. **Click Subjects and then Group.**
4. **Click New and then enter a name for the agent group. For example:** `agentusergroup`
5. **Click OK.**
6. **Under Group, click the agent user group (**`agentusergroup`**).**
7. **On the Edit Group page, click User.**
8. **Select the agent user (**`agentuser`**) under Available and click Add.**
9. **Click Save.**

Assigning Access Privileges to the Agent Group

▼ To Assign Access Privileges to the Agent Group

1. **Login to the OpenSSO Enterprise Administration Console.**
2. **Click Access Control and then the name of the realm where you created the agent group.**

3. **Click Privileges.**
4. **Click the agent profile group. For example:** `agentusergroup`
5. **On the Privileges page, check:**

   `Read and write access to all realm and policy properties`
6. **Click Save.**

Creating the Primary Administrative User in OpenSSO Enterprise

If WebSphere global security is enabled, this user will be able to login to the WebSphere Administration Console.

▼ To Create the Primary Administrative User in OpenSSO Enterprise

1. **Login to the OpenSSO Enterprise Administration Console.**
2. **Click Access Control and then the name of the realm for which you would like to create the primary administrative user.**
3. **Click Subjects and then User.**
4. **Click New and enter values for the following fields:**
   - **ID**: Name of the primary administrative user. For example: `wasadmin`
   - **First Name**: Not required. You can leave this field blank.
   - **Last Name** and **Full Name**: For simplicity, use the same name for each of these values that you specified for ID.
   - **Password** (and confirm): Password for the primary administrative user.
   - **User Status**: Active
5. **Click OK.**

Creating the WebSphere Administrative Group in OpenSSO Enterprise

Any user in this group, in addition to the primary administrative user, can log in to the WebSphere Administration Console.

Note – In the Access Manager 7.1 and Access Manager 7 2005Q4 releases, the user was assigned a specific role. OpenSSO Enterprise uses groups rather than roles for the same functionality.

▼ To Create the WebSphere Administrative Group in OpenSSO Enterprise

1. **Login to the OpenSSO Enterprise Administration Console.**
2. **Click Access Control and then the name of the realm for which you would like to create the WebSphere administrative group.**
3. **Click Subjects and then Group.**
4. **Click New and enter the name for the WebSphere administrative group. For example:** `wasadmingroup`

   Important: Use all lowercase characters for the group name; otherwise, WebSphere might not recognize the name.
5. **Click OK.**
6. **On the returned page, click the WebSphere administrative group (**`wasadmingroup`**).**
7. **Click User.**
8. **Select the primary administrative user (**`wasadmin`**) and any other users you want to be able to log in to the WebSphere Administration Console and click Add.**
9. **Click Save.**

Enabling Cookie Reset for the Agent Profile

▼ To Enable Cookie Reset for the Agent Profile

1. **Log in to the OpenSSO Enterprise Administration Console.**
2. **Click Access Control and then the name of the realm.**
3. **Click Agents, J2EE, and then the name of the agent profile.**
4. **Click SSO.**
5. **Under Cookie Reset:**

- Enable Cookie Reset.
- In the Cookies Reset Name List, add `LtpaToken` and `LtpaToken2`.

  The corresponding configuration properties are:

```
com.sun.identity.agents.config.cookie.reset.enable = true
com.sun.identity.agents.config.cookie.reset.name[0] = LtpaToken
com.sun.identity.agents.config.cookie.reset.name[1] = LtpaToken2
```

These properties are hot–swappable, so you do not need to restart the OpenSSO Enterprise web container instance.

6 **Click Save.**

Specifying the Agent User in the `OpenSSOAgentBootstrap.properties` File

The `OpenSSOAgentBootstrap.properties` file contains the properties required for the agent to start, initialize itself, and connect to the OpenSSO Enterprise server.

The `OpenSSOAgentBootstrap.properties` is located in the following directory on the server where the WebSphere Application Server/Portal Server agent is installed:

Agent-HomeDirectory`/j2ee_agents/websphere_v61_agent/`*agent-instance*`/config`

For example: `/agents/j2ee_agents/websphere_v61_agent/Agent_001/config/`

▼ To Specify the Agent User the `OpenSSOAgentBootstrap.properties` File

1 **Logon to the server where the WebSphere Application Server/Portal Server agent is installed.**

2 **Encrypt the password of the agent user (`agentuser`) that you created in the OpenSSO Enterprise Console under "Creating an Agent User" on page 19.**

   a. **Copy the unencrypted password to the agent profile password file (`wsasagentpw`).**

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

   c. **Encrypt the agent user password using the `agentadmin --encrypt` command following this syntax:**

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

      For example: `# ./agentadmin --encrypt Agent_001 wsasagentpw`

      The command returns the encrypted password. For example:
      `ASEWEJIowNBJHTv1UGD324kmT==`

3 **In the `OpenSSOAgentBootstrap.properties` file, set the following properties:**

- `com.sun.identity.agents.app.username =` *agentuser*
- `com.iplanet.am.service.secret =` *encrypted-agentuser-password*
- `com.sun.identity.agents.config.profilename =` *agentprofile*

where:

- *agentuser* is the agent user you created in the OpenSSO Enterprise Console.
- *encrypted-agentuser-password* is the agent user encrypted password from Step 1.
- *agentprofile* is the WebSphere Application Server/Portal Server agent profile.

4 **Restart the WebSphere Application Server/Portal Server agent web container.**

Performing Global Configuration Tasks for WebSphere Application Server 6.1/7.0

The tasks in this section enable the WebSphere Application Server/Portal Server agent to protect both web applications and the WebSphere Application Server 6.1/7.0 Administration Console.

You perform the following tasks only once for each WebSphere Application Server 6.1/7.0 node, regardless of the number of WebSphere Application Server 6.1/7.0 instances that exist within the node.

- "Setting the WebSphere Application Server 6.1/7.0 Custom Registry" on page 25
- "Adding an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server 6.1/7.0" on page 25
- "Enabling Global Security for WebSphere Application Server 6.1/7.0" on page 26
- "Granting Access to the WebSphere Application Server 6.1/7.0 Administration Console" on page 27
- "Setting the Application Logout URI For the IBM Console" on page 28
- "Installing the Agent Filter for the WebSphere Application Server 6.1/7.0 Administration Console" on page 28
- "Allowing Access to the WebSphere Application Server 6.1/7.0 Administration Console" on page 30
- "Verifying Access to the WebSphere Application Server 6.1/7.0 Administration Console" on page 31

Setting the WebSphere Application Server 6.1/7.0 Custom Registry

▼ To Set the WebSphere Application Server 6.1/7.0 Custom Registry

1. **If needed, start the WebSphere Application Server 6.1/7.0 instance.**
   For example, for an instance named `host1`: `./startServer host1`
2. **Log in to the WebSphere Application Server 6.1/7.0 Administration Console.**
3. **Navigate to the Custom user registry page.**
   a. **Expand the Security node.**
   b. **Click "Secure administration, applications, and infrastructure". A new page opens.**
      For WebSphere Application Server 7.0, click "Global Security".
   c. **Under the Available realm definitions field, select Standalone custom registry.**
   d. **Click the Set as current button.**
   e. **Click the Configure button. A new page opens.**
4. **Set the provider of the custom registry.**
   a. **In the Custom Registry class name field, replace the existing value with the following value:**
      `com.sun.identity.agents.websphere.AmAgentUserRegistry`
   b. **In the "Primary administrative user name" field, enter the administrative user name, which was previously created in OpenSSO Enterprise. For example:** `wasadmin`
   c. **For the Server user identity option, select "Automatically generated server identity".**
   d. **Click OK. The page returns to "Secure administration, applications, and infrastructure".**
      For WebSphere Application Server 7.0, it's "Global Security".
   e. **Click the "Save directly to the master configuration" link.**

Adding an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server 6.1/7.0

This task allows the agent to establish single sign-on (SSO) with the protected WebSphere Application Server 6.1/7.0 instance.

▼ To Add an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server 6.1/7.0

1. **Log in to the WebSphere Application Server 6.1/7.0 Administration Console.**
2. **Navigate to the Interceptors page:**
   a. **Expand the Security node.**
   b. **Click "Secure administration, applications, and infrastructure". A new page opens.**
      For WebSphere Application Server 7.0, click "Global Security".
   c. **Under the "Authentication" option, expand "Web security".**
      For WebSphere Application Server 7.0, expand "Web and SIP Security".
   d. **Click "Trust association". The "Trust association" page opens.**
   e. **Click the Interceptors link. The Interceptors page opens.**
3. **Click New. A new page opens.**
4. **In the "Interceptor class name" field, enter:**
   `com.sun.identity.agents.websphere.AmTrustAssociationInterceptor`
5. **Click OK. The browser returns to the Interceptor page.**
6. **Click "Save directly to the master configuration".**
7. **Navigate again to the Interceptors page by clicking the "Trust association" link at the top of the page.**
8. **Check the "Enable trust association" checkbox.**
9. **Click OK. The page returns to "Secure administration, applications, and infrastructure".**
   For WebSphere Application Server 7.0, it's "Global Security".
10. **Click "Save directly to the master configuration".**

Enabling Global Security for WebSphere Application Server 6.1/7.0

Perform this task only once per WebSphere Application Server 6.1/7.0 node, regardless of the number of WebSphere Application Server 6.1/7.0 instances that exist within the node.

▼ To Enable Global Security for WebSphere Application Server 6.1/7.0

1. **Log in to the WebSphere Application Server 6.1/7.0 Administration Console.**
2. **Navigate to Global security page:**
   a. **Expand the Security node.**
   b. **Click "Secure administration, applications, and infrastructure". A new page opens.**
      For WebSphere Application Server 7.0, click "Global Security".
3. **Check the "Enable administrative security" checkbox.**
4. **Make sure that "Enable application security" is also checked.**
5. **Optionally, enable "Java 2 security" on the same page.**
6. **Click Apply.**
7. **Click "Save directly to the master configuration".**

Granting Access to the WebSphere Application Server 6.1/7.0 Administration Console

▼ To Grant Access to the WebSphere Application Server 6.1/7.0 Administration Console

1. **Log in to the WebSphere Application Server 6.1/7.0 Administration Console.**
2. **Expand the Users and Groups node.**
3. **Click "Administrative Group Roles". A new page opens.**
4. **Click the Add button. A new page opens.**
   a. **In the "Group name" field, enter `wasadmingroup`, which was defined earlier.**
      For WebSphere Application Server 7.0:
      - Select "Map Groups As Specified Below".
      - In the "Search string" field, enter `wasadmingroup`, and then click Search.
      - In the "Available" field, find and select "`id=wasadmingroup,ou=group,`*ROOT_SUFFIX*`@AmRealm`".

- Click the right arrow button to add the results into the "Mapped to role" field.

b. In the "Role(s)" field, select "Administrator".

c. Click OK and return to the "Administrative Group Roles" page.

5 Click "Save directly to the master configuration".

Setting the Application Logout URI For the IBM Console

▼ To Set the Application Logout URI for the IBM Console

1 Log in to the OpenSSO Enterprise Administration Console.

2 Click Access Control and then the name of the realm.

3 Click Agents, J2EE, and then the name of the agent profile.

4 Click Application.

5 Under Application Logout URI, enter the following values:

- **Map Key**: `ibm/console`
- **Corresponding Map Value**: `/ibm/console/logout.do`
- Click Add and then Save.

The corresponding property is: .

```
com.sun.identity.agents.config.logout.uri[ibm/console] = /ibm/console/logout.do
```

This property is hot–swappable, so you do not need to restart the OpenSSO Enterprise web container instance.

Installing the Agent Filter for the WebSphere Application Server 6.1/7.0 Administration Console

The procedures that you have performed up to this point enable the Trust Association Interceptor to protect the Administration Console while users log in and establish the correct principal. However, the Trust Association Interceptor cannot trap logout events or enforce URL policies. The agent filter allows the enforcement of coarse grained URL policies defined within OpenSSO Enterprise to further control the access to protected resources on the WebSphere Application Server 6.1/7.0 Administration Console.

Therefore, you must add the agent filter to the `web.xml` file, as described in the following steps to protect the Administration Console. Without the filter element, you can log in to the Administration Console and perform normal operations, but the logout button will not function.

Note – The agent filter should be the last filter executed in sequence. Therefore, ensure that you insert the agent filter after all other filters in the `web.xml` file.

▼ To Install the Agent Filter for the WebSphere Application Server 6.1/7.0 Administration Console

1 **Locate the `web.xml` file for the WebSphere Application Server 6.1/7.0 instance.**

For example:

DeployContainer-base`/profiles/`*profile name*`/config/cells/cell name/applications/isclite.ear/deployments/isclite/isclite.war/WEB-INF/`

where *DeployContainer-base* represents the directory where the WebSphere Application Server 6.1/7.0 instance was installed.

2 **Back up the `web.xml` file.**

3 **Add the agent filter to the `web.xml` file.**

Ensure that the agent filter you add is the last filter to be executed in sequence. The example shows an excerpt of the `web.xml` file before the agent filter is added:

```
<filter>
     <filter-name>WSCUrlFilter</filter-name>
     <filter-class>com.ibm.ws.console.core.servlet.WSCUrlFilter</filter-class>
</filter>
<filter-mapping>
     <filter-name>WSCUrlFilter</filter-name>
     <servlet-name>action</servlet-name>
</filter-mapping>
<filter-mapping>
    <filter-name>WSCUrlFilter</filter-name>
     <url-pattern>/federatedlogoff</url-pattern>
</filter-mapping>
```

The example shows the agent filter in bold text:

```
<filter>
     <filter-name>WSCUrlFilter</filter-name>
     <filter-class>com.ibm.ws.console.core.servlet.WSCUrlFilter</filter-class>
</filter>
<filter>
```

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

<filter-mapping>
     <filter-name>WSCUrlFilter</filter-name>
     <servlet-name>action</servlet-name>
</filter-mapping>
<filter-mapping>
    <filter-name>WSCUrlFilter</filter-name>
     <url-pattern>/federatedlogoff</url-pattern>
</filter-mapping>

<filter-mapping>
    <filter-name>Agent</filter-name>
    <url-pattern>/*</url-pattern>
 </filter-mapping>
```

4 **Restart the WebSphere Application Server 6.1/7.0 web container instance.**

Allowing Access to the WebSphere Application Server 6.1/7.0 Administration Console

This task involves creating the corresponding URL policies in the OpenSSO Enterprise Console so that a specific user or group has access to the WebSphere Application Server 6.1/7.0 Administration Console.

▼ To Allow Access to the WebSphere Application Server 6.1/7.0 Administration Console

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

2 **Create URL policies that provide the appropriate subjects with access to the WebSphere Application Server 6.1/7.0 Administration Console.**

Ensure that you give access to both HTTP and HTTPS based administration URLs. For example, you might allow the `wasadmingroup` access to the WebSphere Application Server 6.1/7.0 Administration Console by setting the following URL patterns:

- `http://host1.subexample.example.com:9060/*`
- `https://host1.subexample.example.com:9043/*`
- `http://host1.subexample.example.com:9060/*?*`
- `https://host1.subexample.example:9043/*?*`

In this example, the WebSphere Application Server 6.1/7.0 Administration Console is running with the HTTP protocol on port 9060 and the HTTPS protocol on port 9043. All other changes

to the agent configuration to trap logout events have already been configured by the agent installer. Note that the agent is configured in the most restrictive mode `ALL` at this point.

Verifying Access to the WebSphere Application Server 6.1/7.0 Administration Console

This task involves verifying that the previous tasks were performed properly and that the subjects assigned access to the WebSphere Application Server 6.1/7.0 Administration Console can access the console.

▼ To Verify Access to the WebSphere Application Server 6.1/7.0 Administration Console

1. **Start the WebSphere Application Server 6.1/7.0 instance that you just configured.**
2. **Run the WebSphere Application Server 6.1/7.0 agent in message mode.**
3. **Log in to the WebSphere Application Server 6.1/7.0 Administration Console as a user who belongs to the WebSphere administrative group. For example:** `wasadmin`
4. **If the user is properly redirected to OpenSSO Enterprise, enter the user name and password for any user assigned to the** `wasadmingroup` **in OpenSSO Enterprise.**

   If you are allowed access, you should be redirected to the WebSphere Application Server 6.1/7.0 Administration Console.
5. **Perform normal operations in the WebSphere Application Server 6.1/7.0 Administration Console.**
6. **Click logout.**

   You should be redirected to the OpenSSO Enterprise Console.

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** `websphere_v61_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 WebSphere Application Server 6.1/7.0 instance using the WebSphere Application Server 6.1/7.0 administration console or deployment command.**

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 Applications Protected by the WebSphere Application Server/Portal Server Agent

Installing the Agent Filter for a Deployed Application on the WebSphere Application Server/Portal Server Agent

To install the agent filter, modify the deployment descriptor of each application that you want to protect.

▼ To Install the Agent Filter for the WebSphere Application Server/Portal Server Agent

1 **Ensure that the application you want to protect is not currently deployed on WebSphere Application Server 6.1/7.0.**

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: WebSphere Application Server/Portal Server 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 Deploy (or redeploy) the application on the WebSphere Application Server 6.1/7.0 web container.

The agent filter is then added to the application.

Next Steps 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 37.

Optional Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent

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 WebSphere Application Server/Portal Server 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 wsasagentpw
      ```

      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 WebSphere Application Server 6.1/7.0 instance that is being protected by the policy agent.**

Creating the Necessary URL Policies

If the WebSphere Application Server/Portal Server agent is configured to operate in the `URL_POLICY` or `ALL` filter mode, you must create the appropriate URL policies. For instance, if

WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server agent.

For information about how to create these policies using the OpenSSO Enterprise Console or command-line utilities, see the *Sun OpenSSO Enterprise 8.0 Administration Guide*.

Configuring Web Services Security for the WebSphere Application Server/Portal Server Agent

The WebSphere Application Server/Portal Server agent supports Web Services Security (WSS) for web service providers. A web service provider (WSP) deployed on WebSphere Application Server 6.1/7.0 protected by the agent can have additional security provided by the agent. For example, you can configure the WebSphere Application Server/Portal Server agent and OpenSSO Enterprise server to support various Web Services Security profiles, including Username token, X509 token, and SAML2 token.

Configuring the WebSphere Application Server/Portal Server agent to use Web Services Security with OpenSSO Enterprise is similar to configuring other Java EE policy agents. For information and the general configuration steps, see "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*.

In addition to the general steps, perform the following additional steps depending on the version of WebSphere Application Server you are using:

- "Configuring Web Services Security on WebSphere Application Server 6.1" on page 36
- "Configuring Web Services Security on WebSphere Application Server 7.0" on page 36

Configuring Web Services Security on WebSphere Application Server 6.1

▼ To Configure Web Services Security on WebSphere Application Server 6.1

1. **Perform the general steps, as described 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*.**
2. **Stop WebSphere Application Server 6.1.**
3. **Install the WebSphere Application Server 6.1 Feature Pack for Web Services onto WebSphere Application Server 6.1.**

   For information, see `http://www-01.ibm.com/software/webservers/appserv/was/featurepacks/`.
4. **Copy the `xmlsec.jar`, `xercesImpl.jar` and `xalan.jar` files from the OpenSSO Enterprise server deployment to the *WebSphereInstallDirectory*`/AppServer/lib/ext` directory.**

   For example: `/opt/IBM/WebSphere/AppServer/lib/ext`
5. **Download `bcprov-jdk15-141.jar` from `http://bouncycastle.org` and copy it to the *WebSphereInstallDirectory*`/AppServer/java/jre/lib/ext` directory.**
6. **Add the Bouncy Castle provider to the *WebSphereInstallDirectory*`/AppServer/java/jre/lib/security/java.security` file. For example:**

   ```
   security.provider.9=org.bouncycastle.jce.provider.BouncyCastleProvider
   ```

   Change the provider number accordingly.
7. **Start WebSphere Application Server 6.1**

Configuring Web Services Security on WebSphere Application Server 7.0

▼ To Configure Web Services Security on WebSphere Application Server 7.0

1. **Perform the general steps, as described 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*.**
2. **Stop WebSphere Application Server 7.0.**

3. **Copy the `xmlsec.jar`, `xercesImpl.jar`, and `xalan.jar` files from the OpenSSO Enterprise server deployment to the *WebSphereInstallDirectory*`/AppServer/lib/ext` directory.**

   For example: `/opt/IBM/WebSphere/AppServer/lib/ext`

4. **Start WebSphere Application Server 7.0.**

Deploying the Java EE Policy Agent Sample Application

Deploying the policy agent sample application is optional. However. after you install the WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server agent, the sample application is available as:

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

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

Installing and Configuring the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

Installing the WebSphere Application Server/Portal Server agent in a Network Deployment environment is similar to the installation process for an environment with a single Application Server instance. However, you must also install and configure an agent instance onto the Deployment Manager server instance, each Node Agent instance, and each Application Server instance. The Application Server instance might be or might not be within a cluster.

- "Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment" on page 38

- "Installing the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment" on page 38
- "Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment" on page 42

Caution – Before you install and configure the WebSphere Application Server/Portal Server agent, the Network Deployment environment must already be setup properly. This guide does not cover installing or configuring the Network Deployment environment itself. Each server instance's configuration should also be synchronized with its corresponding part in the Deployment Manager's profile. That is, each server instance's `server.xml` file in the remote host should be the same as the corresponding copy in the Deployment Manager's profile. One way to achieve this synchronization is to run the `syncNode.sh` (or `syncNode.bat` on Windows) command on each node for each profile.

You must also stop the Network Deployment, including the Deployment Manager server instance, all Node Agent instances, and all Application Server instances.

Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

The pre-installation tasks are the same as "Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent" on page 6.

Note: Each agent instance should have a unique agent profile. You can create each agent profile as described in "Creating an Agent Profile" on page 8 or during the agent installation using the `agentadmin --custom-install` option.

Installing the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

The following install sequence is recommended for a Network Deployment environment, although it is not necessarily in a required order:

- "Installing the WebSphere Application Server/Portal Server Agent on the Deployment Manager Instance" on page 39
- "Installing WebSphere Application Server/Portal Server Agent on Each Node Agent" on page 40
- "Installing the WebSphere Application Server/Portal Server Agent on Each Application Server Instance" on page 41

Installing the WebSphere Application Server/Portal Server Agent on the Deployment Manager Instance

Install the first instance of the WebSphere Application Server/Portal Server agent on the Deployment Manager instance.

▼ To the WebSphere Application Server/Portal Server Agent on the Deployment Manager Instance

1. **Ensure that the WebSphere Application Server 6.1 or 7.0 Network Deployment is down.**

2. **On the machine running Deployment Manager, install the agent onto the Deployment Manager server instance, as described in "Installing the WebSphere Application Server/Portal Server Agent" on page 11.**

   Installation considerations are:

   - Use the `agentadmin --custom-install` option.
   - Several prompts specific to this installation are:

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server instance. For example: `/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/`*hostname*`Cell01/nodes/`*hostname*`CellManager01/servers/dmgr`
Server Instance name	Name of the WebSphere Application Server instance. For example: `dmgr`
Agent URL	Agent URL, including the deployment URIs. For example: `http://agenthost.example.com:9080/agentapp` 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. For more information, see "Deploying the Agent Application" on page 31. **Note**: Since the `agentapp` cannot be deployed onto the Deployment Manager instance, this URL can point to an Application Server instance on the same host with the `agentapp` deployed.

Installing WebSphere Application Server/Portal Server Agent on Each Node Agent

▼ To Installing WebSphere Application Server/Portal Server Agent on Each Node Agent

1 **Ensure that the WebSphere Application Server 6.1 or 7.0 Network Deployment is down.**

2 **On the machine running the Node Agent, install the agent onto the Node Agent instance as, described in "Installing the WebSphere Application Server/Portal Server Agent" on page 11.**

Installation considerations are:

- Use the `agentadmin --custom-install` option.
- Several prompts specific to this installation are:

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server instance. For example: `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/`*hostname*`Cell01/nodes/`*hostname*`Node01/servers/nodeagent`
Server Instance name	Name of the WebSphere Application Server instance. For example: `nodeagent`
Agent URL	Agent URL, including the deployment URIs. For example: `http://agenthost.example.com:9080/agentapp` 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. For more information, see "Deploying the Agent Application" on page 31. **Note**: Since the `agentapp` cannot be deployed onto the Node Agent instance, this URL can point to an Application Server instance on the same host with the `agentapp` deployed.

3 **Copy the Node Agent's `server.xml` file to overwrite its corresponding copy under the Deployment Manager's profile.**

For example, copy:

`/opt/IBM/WebSphere/AppServer/profiles/`**`AppSrv01`**`/config/cells/`
hostname`Cell01/nodes/`*hostname*`Node01/servers/nodeagent/server.xml`

to overwrite:

```
/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/
hostnameCell01/nodes/hostnameNode01/servers/nodeagent/server.xml
```

Note: The above two `server.xml` files should be synchronized before installation, so this copy operation will not cause a mismatch. Otherwise you must find out the changes in `server.xml` (compared with original copy with a name such as `server.xml-preAmAgent-`*timestamp*) by the agent installer and merge the changes with its corresponding copy in the Deployment Manager's profile.

If the Node Agent is on a remote host from the Deployment Manager, its `server.xml` file on the remote host should be copied or FTPed to the host of the Deployment Manager and overwrite its own corresponding copy in the Deployment Manager profile as above.

Caution: Each Node Agent has its own copy of `server.xml` in Deployment Manager, and overwriting a file mistakenly can cause the other server instances to malfunction.

Installing the WebSphere Application Server/Portal Server Agent on Each Application Server Instance

▼ To Install the WebSphere Application Server/Portal Server Agent on Each Application Server Instance

1. **Ensure that the WebSphere Application Server 6.1 or 7.0 Network Deployment is down.**

2. **On the machine running the Application Server instances, install the WebSphere Application Server/Portal Server agent onto each Application Server instance, as described in "Installing the WebSphere Application Server/Portal Server Agent" on page 11.**

   Installation considerations are:

   - Use the `agentadmin --custom-install` option.
   - Several prompts specific to this installation are:

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server instance. For example: `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/` *hostname*`Cell01/nodes/`*hostname*`Node01/servers/server1`
Server Instance name	Name of the WebSphere Application Server instance. For example: `server1`

Prompt	Description
Agent URL	Agent URL, including the deployment URIs. For example: `http://agenthost.example.com:9080/agentapp` 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. For more information, see "Deploying the Agent Application" on page 31.

3 **Copy the Application Server `server1` `server.xml` file to overwrite its corresponding copy under the Deployment Manager's profile.**

For example, copy:

`/opt/IBM/WebSphere/AppServer/profiles/`**`AppSrv01`**`/config/cells/`
hostname`Cell01/nodes/`*hostname*`Node01/servers/server1/server.xml`

to overwrite:

`/opt/IBM/WebSphere/AppServer/profiles/`**`Dmgr01`**`/config/cells/`
hostname`Cell01/nodes/`*hostname*`Node01/servers/server1/server.xml`

Note: The above two `server.xml` files should be synchronized before installation, so this copy operation will not cause a mismatch. Otherwise you must find out the changes in `server.xml` (compared with original copy with a name such as `server.xml-preAmAgent-`*timestamp*) by the agent installer and merge the changes with its corresponding copy in the Deployment Manager's profile.

If the Node Agent is on a remote host from the Deployment Manager, its `server.xml` file on the remote host should be copied or FTPed to the host of the Deployment Manager and overwrite its own corresponding copy in the Deployment Manager profile as above.

Caution: Each Application Server instance has its own copy of `server.xml` in Deployment Manager, and overwriting a file mistakenly can cause the other server instances to malfunction.

Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

▼ To Perform Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

1 **Perform the following steps once to configure all agent instances.**

- "Creating an Agent User" on page 19

- "Creating a New Agent Group for the Agent User" on page 20
- "Assigning Access Privileges to the Agent Group" on page 20
- "Creating the Primary Administrative User in OpenSSO Enterprise" on page 21
- "Creating the WebSphere Administrative Group in OpenSSO Enterprise" on page 21

2 **Perform "Enabling Cookie Reset for the Agent Profile" on page 22"Enabling Cookie Reset for the Agent Profile" on page 22 for each agent profile, including the agent profile for the Deployment Manager, Node Agent, or Application Server instances.**

3 **Perform "Specifying the Agent User in the `OpenSSOAgentBootstrap.properties` File" on page 23 for each agent instance's `OpenSSOAgentBootstrap.properties` file.**

Note: All agent instances can share the same agent user, but each agent instance still keeps its unique agent profile using the following property:

```
com.sun.identity.agents.config.profilename = agent-profile-name
```

4 **In "Performing Global Configuration Tasks for WebSphere Application Server 6.1/7.0" on page 24, perform each task once for the configuration of all agent instances.**

5 **Perform "Deploying the Agent Application" on page 31 for each WebSphere Application Server instance.**

6 **For "Configuring Applications Protected by the WebSphere Application Server/Portal Server Agent" on page 32, perform this task for each application to be protected by the WebSphere Application Server/Portal Server agent.**

7 **Optionally, consider the "Optional Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent" on page 33.**

Installing and Configuring the WebSphere Application Server/Portal Server Agent on WebSphere Portal Server 6.1

This section describes how to install and configure the WebSphere Application Server/Portal Server agent in a single WebSphere Portal Server 6.1 environment, including:

- "Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent in Single WebSphere Portal Server 6.1 Environment" on page 44
- "Installing the WebSphere Application Server/Portal Server Agent in a Single WebSphere Portal Server 6.1 Environment" on page 44
- "Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Single WebSphere Portal Server 6.1 Environment" on page 46

To install this agent in a Network Deployment environment, you must install and configure one agent instance onto the Deployment Manager server instance, each Node Agent instance, each Portal Server instance, and each Application Server instance.

Caution – Before installing the agent, you must stop the WebSphere Portal Server 6.1 environment, including the Application Server `server1` instance and the Portal Server `WebSphere_Portal` instance.

Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent in Single WebSphere Portal Server 6.1 Environment

The pre-installation tasks for WebSphere Portal Server 6.1 are the same as "Pre-Installation Tasks for the WebSphere Application Server/Portal Server Agent" on page 6.

Note: Each agent instance must have a unique agent profile. You can create each agent profile as described in "Creating an Agent Profile" on page 8 or during the agent installation using the `agentadmin --custom-install` option.

Installing the WebSphere Application Server/Portal Server Agent in a Single WebSphere Portal Server 6.1 Environment

When you install the agent on WebSphere Portal Server 6.1, you must run the agent installation program on every instance of the underlying WebSphere Application Server. In a single Portal Server environment, this includes two instances: the default instance, often named `server1`, and the WebSphere Portal Server 6.1 instance, often named `WebSphere_Portal`.

In a single Portal Server environment, the recommended installation sequence is:

- "Installing the WebSphere Application Server/Portal Server Agent on the Application Server `server1` Instance" on page 44
- "Installing the WebSphere Application Server/Portal Server Agent on the WebSphere Portal Server 6.1 `WebSphere_Portal` Instance" on page 45

Installing the WebSphere Application Server/Portal Server Agent on the Application Server `server1` Instance

Install the first instance of the WebSphere Application Server/Portal Server agent on the Application Server `server1` instance.

▼ To Install the WebSphere Application Server/Portal Server Agent on the Application Server `server1` Instance

1. **Ensure that the WebSphere Portal Server 6.1 environment is down.**

2. **On the machine running WebSphere Portal Server 6.1, install the agent onto the Application Server `server1` instance, as described in "Installing the WebSphere Application Server/Portal Server Agent" on page 11.**

   Installation considerations are:

   - Use the `agentadmin --custom-install` option.
   - Several prompts specific to this installation are:

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server instance. For example: `/opt/IBM/WebSphere/wp_profile/config/cells/`*hostname*`/nodes/`*hostname*`/serv`
Server Instance Name	Name of the WebSphere Application Server instance. For example: `server1`
Agent URL	Agent URL, including the deployment URI. For example: `http://agenthost.example.com:10000/agentapp` **Note**: This URL is where `agentapp` will be deployed. 10000 is default port of the `server1` instance.

Installing the WebSphere Application Server/Portal Server Agent on the WebSphere Portal Server 6.1 `WebSphere_Portal` Instance

▼ To Install the WebSphere Application Server/Portal Server Agent on the WebSphere Portal Server 6.1 `WebSphere_Portal` Instance

1. **Ensure that the WebSphere Portal Server 6.1 environment is down.**

2. **On the machine running WebSphere Portal Server 6.1, install the agent onto the Portal Server `WebSphere_Portal` instance, as described in "Installing the WebSphere Application Server/Portal Server Agent" on page 11.**

   Installation considerations are:

   - Use the `agentadmin --custom-install` option.
   - Several prompts specific to this installation are:

Prompt	Description
Instance Config Directory	Path to the configuration directory for the WebSphere Application Server instance. For example: `/opt/IBM/WebSphere/wp_profile/config/cells/`*hostname*`/nodes/`*hostname*`/servers/We`
Server Instance Name	Name of the WebSphere Application Server instance. For example: `WebSphere_Portal`
Agent URL	Agent URL, including the deployment URI. For example: `http://agenthost.example.com:10040/agentapp` **Note**: This URL is where `agentapp` will be deployed. 10040 is default port of the `WebSphere_Portal` instance.

Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent in a Single WebSphere Portal Server 6.1 Environment

Some of the following post-installation tasks are unique to WebSphere Portal Server 6.1, while other tasks are identical to the same task for WebSphere Application Server:

WebSphere Portal Server: Creating the Primary Administrative User in OpenSSO Enterprise

Perform this task once for all agent instances. This user (for example, `wasadmin`) is either the administrative user who installs WebSphere Portal Server or an administrative user designated after the WebSphere Portal Server installation is finished.

Note: You can skip this task if this administrative user or an equivalent has already been configured to authenticate with OpenSSO Enterprise.

Otherwise, by default, create `wasadmin` in the OpenSSO embedded Configuration Data Store. This data store needs to be involved in authentication with OpenSSO Enterprise (for example, via an authentication chain).

Follow the steps in "Creating the Primary Administrative User in OpenSSO Enterprise" on page 21.

WebSphere Portal Server: Deploying the Agent Application

Perform this task for each WebSphere Application Server instance, including the Application Server `server1` instance and the Portal Server `WebSphere_Portal` instance.

Follow the steps in "Deploying the Agent Application" on page 31.

WebSphere Portal Server: Performing Global Configuration Tasks

Perform the following tasks **only** if you are also "Performing Global Configuration Tasks for WebSphere Application Server 6.1/7.0" on page 24:

- "WebSphere Portal Server: Adding an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server" on page 47
- "WebSphere Portal Server: Changing the Logout Link Actions for WebSphere Portal Server 6.1" on page 47
- "WebSphere Portal Server: Enabling Global Security for WebSphere Application Server" on page 48
- "WebSphere Portal Server: Setting the Application Logout URI For the IBM Console" on page 48
- "WebSphere Portal Server: Enabling Cookie Reset for the Agent Profile" on page 49
- "WebSphere Portal Server: Installing the Agent Filter for the WebSphere Application Server Administration Console" on page 49

WebSphere Portal Server: Adding an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server

Follow the steps in "Adding an OpenSSO Enterprise Trust Association Interceptor to WebSphere Application Server 6.1/7.0" on page 25.

WebSphere Portal Server: Changing the Logout Link Actions for WebSphere Portal Server 6.1

This task provides a seamless user experience of single sign-off with OpenSSO Enterprise.

To Change the Logout Link Actions for WebSphere Portal Server 6.1

1. Ensure that the WebSphere Application Server and WebSphere Portal Server 6.1 instances are running.
2. Access the WebSphere administrative console by entering the following URL in the location field of a Web browser:

   `http://example.com:`*admin_port*`/ibm/console`

where `example.com` is the name of the server and *admin_port* is the port assigned to the administrative console.

3. Click Resources > Resources Environment > Resource Environment Providers.
4. On the Resource Environment Providers page, make the appropriate selection, depending on your version of WebSphere Application Server and your portal environment:
   - For WebSphere Application Server Version 6.1, select the appropriate node or cluster from the scopes pull-down list, depending on your portal environment.
   - For WebSphere Application Server Version 7.0, select the appropriate node or cluster from the scopes pull-down list. Or uncheck the Show Scope selection drop-down checkbox and select one of the following options, depending on your portal environment:
     - If your portal is running as a single server, select Browse Nodes and select the node.
     - If your portal is installed in a cluster, select Browse Clusters and select the portal cluster.
5. Select the "WP ConfigService" service.
6. Click Custom Properties.
7. Do the following, as required:
   - Set `redirect.logout` to `true`.
   - Set `redirect.logout.ssl` to `true` or `false`, depending upon the environment.
   - Set `redirect.logout.url` to the OpenSSO Enterprise logout URL. For example:

     ```
     http://opensso-host.example.com:8080/opensso/UI/Logout
     ```
   - When you are done, click Save at the top of the screen under Message(s).
8. If you are running a cluster configuration, replicate your changes to the cluster.

WebSphere Portal Server: Enabling Global Security for WebSphere Application Server

If Global Security is not enabled, follow the steps in "Enabling Global Security for WebSphere Application Server 6.1/7.0" on page 26.

WebSphere Portal Server: Setting the Application Logout URI For the IBM Console

For each agent profile, including the agent profile for the WebSphere Application Server `server1` instance and the WebSphere Portal Server `WebSphere_Portal` instance, perform the steps in "Setting the Application Logout URI For the IBM Console" on page 28.

WebSphere Portal Server: Enabling Cookie Reset for the Agent Profile

For each agent profile, including the agent profile for the WebSphere Application Server `server1` instance and the WebSphere Portal Server `WebSphere_Portal` instance, perform the steps in "Enabling Cookie Reset for the Agent Profile" on page 22.

WebSphere Portal Server: Installing the Agent Filter for the WebSphere Application Server Administration Console

Perform the steps in "Installing the Agent Filter for the WebSphere Application Server 6.1/7.0 Administration Console" on page 28.

Adding the Agent Filter to the WebSphere Portal Server 6.1 Application

This required task integrates the WebSphere Portal Server 6.1 instance with the OpenSSO Enterprise environment.

Note: Perform this task only once per WebSphere Portal Server 6.1 instance for a given host.

The WebSphere Application Server/Portal Server agent provides a `servlet` filter that you can add to the WebSphere Portal Server 6.1 application. This filter allows the enforcement of coarse grained URL policies defined within OpenSSO Enterprise server to further control the access to protected resources on the WebSphere Portal Server 6.1 instance. The filter can also be configured to provide additional personalization information in the form of HTTP headers, cookies, or HTTP request attributes that can be used to further enhance the functionality of the protected components.

▼ To Add the Agent Filter to the WebSphere Portal Server 6.1 Application

1. **Ensure that the WebSphere Portal Server 6.1 environment is down.**

2. **Locate the `wps.war/WEB-INF/web.xml` file, which contains the deployment descriptors for WebSphere Portal Server 6.1.**

   WebSphere Application Server can read this file at runtime from either of the following directories:

   - *WAS-base*`/wp_profile/installedApps/`*Cell-Name*`/wps.ear/wps.war/WEB-INF`
   - *WAS-base*`/wp_profile/config/cells/`*Cell-Name*`/applications/wps.ear/deployments/wps/wp`

   where:

   - *WAS-base* represents the directory where WebSphere Portal Server 6.1 was installed
   - *Cell-Name* represents the WebSphere Portal Server 6.1 cell protected by the agent. The default is `hostname`.

3 **Backup the two `web.xml` files before modifying the deployment descriptors.**

Since you will modify the deployment descriptor in the next step, creating backup files is important, especially if you need to uninstall the agent in the future.

4 **Edit both `web.xml` files from the previous step, as follows:**

```
<display-name>WebSphere Portal Server</display-name>

<filter id="Filter_PolicyAgent">
<filter-name>Policy Agent</filter-name>
<filter-class>
  com.sun.identity.agents.filter.AmAgentFilter
</filter-class>
</filter>

... //other filter definitions

<filter-mapping id="FilterMapping_PolicyAgent">
  <filter-name>Policy Agent</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

... //other filter mappings

</web-app>
```

WebSphere Portal Server: Creating the Necessary URL Policies

If the WebSphere Application Server/Portal Server agent is installed and configured to operate in ALL mode, you must create the appropriate URL policies.

Note: Since WebSphere Portal Server is protected by J2EE declarative security, the agent should operate in J2EE_POLICY or ALL mode.

For example, if WebSphere Application Server with the Administration Console is listening on ports 10027 (`http`) and 10041 (`https`), respectively, and WebSphere Portal Server is listening on port 10040 (`http`), create the following polices for the WebSphere Administrative user ID (`wasadmin` or `wpsadmin`) to allow the user access to the WebSphere Administration Console and Portal Server URLs:

- "URLs for the Portal Server `WebSphere_Portal` Instance" on page 50
- "URLs for the Application Server `server1` Instance" on page 51

URLs for the Portal Server `WebSphere_Portal` Instance

- `http://agenthost.example.com:10027/*`
- `https://agenthost.example.com:10041/*`
- `http://agenthost.example.com:10040/*`

- `https://agenthost.example.com:10041/*?*`
- `http://agenthost.example.com:10040/*?*`

Notes:

- These examples assume that `http://agenthost.example.com:10027/ibm/console` is the Administration console URL on `WebSphere_Portal` and `http://agenthost.example.com:10040/wps/myportal` is the Portal Server URL.
- Port 10041 is the corresponding `https` port of `http` port 10027. When an `http` request comes to port 10027, it will be redirected to 10041 as an `https` request.
- Only the protected portal `http://agenthost.example.com:10040/wps/myportal` is supported by the agent. The non-protected portal `http://agenthost.example.com:10040/wps/portal` is not supported.

URLs for the Application Server `server1` Instance

- `http://agenthost.example.com:10001/*`
- `https://agenthost.example.com:10003/*`
- `http://agenthost.example.com:10000/*`
- `https://agenthost.example.com:10003/*?*`
- `http://agenthost.example.com:10000/*?*`

Notes:

- These examples assume that `http://agenthost.example.com:10001/ibm/console` is the Administration console URL on `server1`and `http://agenthost.example.com:10000` is the `server1` server URL.
- Port 10003 is the corresponding `https` port of `http` port 10001. When an `http` request comes to port 10001, it will be redirected to 10003 as an `https` request.

WebSphere Portal Server: Considering Optional Tasks

Consider the other "Optional Post-Installation Tasks for the WebSphere Application Server/Portal Server Agent" on page 33.

WebSphere Portal Server: Restarting WebSphere Portal Server 6.1

After you are finished performing all post-installation tasks, restart the WebSphere Portal Server 6.1 environment.

Managing the WebSphere Application Server/Portal Server 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.

The following property in the OpenSSO Enterprise server Agent Service schema (`AgentService.xml` file) indicates that the configuration is local:

```
com.sun.identity.agents.config.repository.location=local
```

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 WebSphere Application Server/Portal Server Agent

Preparing to Uninstall the WebSphere Application Server/Portal Server Agent

▼ To Prepare to Uninstall WebSphere Application Server/Portal Server Agent

1. **Undeploy any applications protected by the WebSphere Application Server/Portal Server 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 WebSphere Application Server/Portal Server 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 WebSphere Application Server 6.1/7.0 instances that is being protected by the agent is stopped.**

Uninstalling the WebSphere Application Server/Portal Server Agent Using the `agentadmin` Program

▼ To Uninstall the WebSphere Application Server/Portal Server 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 following values for the WebSphere Application Server 6.1/7.0 instance you are unistalling:**

   - Configuration directory path
   - Server instance name
   - Install root directory

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 WebSphere Application Server/Portal Server Agent

```
************************************************************************
Welcome to the OpenSSO Policy Agent for IBM WebSphere Application Server 6.1
************************************************************************

Enter the fully qualified path to the configuration directory of the Server
Instance for the WebSphere node.
[ ? : Help, ! : Exit ]
Enter the Instance Config Directory
[/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
  Node01Cell/nodes/Node01/servers/server1]:
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
  agenthostNode01Cell/nodes/agenthostNode01/servers/server1

Enter the Server Instance name.
```

```
[ ? : Help, < : Back, ! : Exit ]
Enter the Server Instance name [server1]:

Enter the WebSphere Install Root directory.
[ ? : Help, < : Back, ! : Exit ]
Enter the WebSphere Install Root directory
[/opt/IBM/WebSphere/AppServer]:

---------------------------------------------
SUMMARY OF YOUR RESPONSES
---------------------------------------------
Instance Config Directory :
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/
  agenthostNode01Cell/nodes/agenthostNode01/servers/server1
Instance Server name : server1
WebSphere Install Root Directory : /opt/IBM/WebSphere/AppServer
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]:
Remove agent.jar,openssoclientsdk.jar from
/opt/IBM/WebSphere/AppServer/lib/ext...DONE.
Deleting the config directory
/agents/j2ee_agents/websphere_v61_agent/Agent_001/config
...DONE.
Unconfigure server.xml file
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/Node01Cell/nodes/
Node01/servers/server1/server.xml
...DONE.

Uninstall log file location:
/agents/j2ee_agents/websphere_v61_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.

Uninstalling the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment

To remove each WebSphere Application Server/Portal Server agent instance, follow the steps in "Uninstalling the WebSphere Application Server/Portal Server Agent" on page 53. After running `agentadmin --uninstall`, copy each server instance's `server.xml` file to overwrite its corresponding copy in the Deployment Manager's profile, as described in "Installing and Configuring the WebSphere Application Server/Portal Server Agent in a Network Deployment Environment" on page 37.

Migrating a Version 2.2 WebSphere Application Server 6.1/7.0 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 WebSphere Application Server agent, download the version 3.0 WebSphere Application Server/Portal Server 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 WSASv3Agent 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 33.

▼ 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 WebSphere Application Server 6.1/7.0 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 WebSphere Application Server/Portal Server agent in the `v30agent` directory:

   ```
   cd /v30agent/j2ee_agents/websphere_v61_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/websphere_v61_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 = WSASv3Agent
```

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 WSASv3Agent -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 WSASv3Agent` 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 WebSphere Application Server 6.1/7.0 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 WebSphere Application Server/Portal Server Agent" on page 52.

Sun Related Information

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 IBM WebSphere Application Server 6.1/7.0 and WebSphere Portal Server 6.1*, and the part number is 820-7250.

Revision History

Part Number	Date	Description
820–7250–11	July 1, 2009	Added support for WebSphere Portal Server 6.1.
820–7250–10	April 22, 2009	Initial release.