Sun OpenSSO Enterprise 8.0 Upgrade Guide

Part No: 820–5019–14
March 9, 2010

100310@23474

OpenSSO Enterprise 8.0 Upgrade Guide

Last updated March 9, 2010

The *Sun OpenSSO Enterprise 8.0 Upgrade Guide* describes how to upgrade Sun Java System Access Manager and Sun Java System Federation Manager to Sun OpenSSO Enterprise 8.0. The upgrade process involves upgrading an existing Access Manager or Federation Manager server instance and the corresponding configuration data stored in Sun Java System Directory Server.

Contents

Planning Your OpenSSO Enterprise 8.0 Upgrade

Upgrading to Sun OpenSSO Enterprise 8.0 is supported from the following releases and platforms:

Previous Release	Upgrade Supported From This Platform
Sun Java System Access Manager 7.1 Upgrade is supported for: ■ Sun Java Enterprise System package based installations ■ WAR file deployment only if the configuration data is in Sun Java System Directory Server.	Solaris SPARC, Solaris x86, Linux, and Windows systems
Sun Java System Access Manager 7 2005Q4	Solaris SPARC, Solaris x86, and Linux systems
Sun Java System Federation Manager 7.0	Solaris SPARC, Solaris x86, Linux, and Windows systems

Caution – Upgrade of the configuration data is supported only from and to Sun Java System Directory Server. If the configuration data for an Access Manager 7.1 WAR file deployment is stored using the flat file system, upgrade to OpenSSO Enterprise 8.0 is not supported.

Upgrade is **not** supported for the following separately installed features:

- Access Manager or Federation Manager AMSDK
- Access Manager or Federation Manager client SDK
- Distributed Authentication UI server. See "Considerations for a Distributed Authentication UI Server" on page 5.
- IDP Discovery Service
- Remote console

Additional information is in the following sections.

Considerations for OpenSSO Enterprise 8.0 Patch Releases

Sun periodically releases patches for OpenSSO Enterprise 8.0 on `http://sunsolve.sun.com/`. To find the latest patch, search for patch ID 141655.

To migrate a deployment from Access Manager 7.1 or Access Manager 7 2005Q4 to an OpenSSO Enterprise 8.0 patch release, follow these general steps:

1. Upgrade to OpenSSO Enterprise 8.0, as described in this guide.
2. Apply the patch release, as described in Chapter 23, "Patching OpenSSO Enterprise 8.0," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

Note – Always run the latest versions of the `ssoupgrade` or `ssoupgrade.bat` script, `ssopatch` or `ssopatch.bat` utility, and `updateschema` or `updateschema.bat` script from the OpenSSO Enterprise 8.0 patch release.

Upgrade Considerations for Legacy and Realm Mode

The following Legacy and Realm mode upgrades are supported:

- Legacy to Legacy mode
- Legacy to Realm mode
- Realm to Realm mode

Considerations for a Distributed Authentication UI Server

Upgrade is not supported for an Access Manager 7.x Distributed Authentication UI server deployment. To move from an Access Manager 7.x deployment, you must remove the old deployment and then install the OpenSSO Enterprise 8.0 version, as described in Chapter 9, "Deploying a Distributed Authentication UI Server," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

OpenSSO Enterprise 8.0 can coexist with an Access Manager 7.1 Distributed Authentication UI server deployment. However, make sure that the following properties in the `AMConfig.properties` file on the Distributed Authentication UI server side in the `WEB-INF` directory are in sync with the OpenSSO Enterprise 8.0 server instance:

- `com.iplanet.am.naming.url`
- `com.iplanet.am.server.protocol`
- `com.iplanet.am.server.host`
- `com.iplanet.am.server.port`
- `com.sun.identity.agents.app.username`
- `com.iplanet.am.service.password`

Note: If you make any changes to the `AMconfig.properties` file, you must restart the Distributed Authentication UI server.

Coexistence with OpenSSO Enterprise 8.0

Coexistence can occur when instances of OpenSSO Enterprise and Access Manager 7.1 access the same Directory Server schema. (In previous versions of Access Manager, the Directory Server schema contains the server's configuration data.) Thus, OpenSSO Enterprise 8.0 can coexist with an instance of Access Manager 7.1 only if the older version was installed with the Directory Server schema. Coexistence mode denotes that customer has executed the `ssopreupgrade` script to remove the packages but not the `ssoupgrade` script to update the schema.

Coexistence usually occurs when multiple instances of Access Manager 7.1, accessing the same Directory Server schema, are being upgraded sequentially, one instance at a time. OpenSSO Enterprise 8.0 will continue to work with the Access Manager 7.1 schema and support all Access Manager 7.1 features (except for the Liberty ID-FF metadata as described in "Backward Compatibility with OpenSSO Enterprise 8.0" on page 6) until the schema is upgraded.

Important. In coexistence mode, all Access Manager 7.1 instances accessing the same Directory Server schema must have the same deployment URI (for example `/amserver`).

Coexistence is **not** supported between OpenSSO Enterprise 8.0 server and these releases:

- Access Manager 7 2005Q4
- Federation Manager 7.0

More information about upgrading multiple instances of Access Manager is in "Upgrading Multiple Instances of Access Manager" on page 25.

Tip – Upgrading from older versions of Access Manager might cause issues when logging in and accessing realms in coexistence mode. There is no current workaround for this issue. It is suggested that you upgrade to OpenSSO Enterprise 8.0 update 1 once it is available.

Backward Compatibility with OpenSSO Enterprise 8.0

Backward compatibility is supported for all Access Manager 7.1 and Access Manager 7 2005Q4 existing features including the full SDK and the client SDK APIs. Backward compatibility is **not** supported for:

- Access Manager 6 2005Q1 (6.3) and earlier releases
- Liberty ID-FF schema metadata: Liberty ID-FF profiles do not work unless you upgrade the Access Manager or Federation Manager schema in Directory Server.

OpenSSO Enterprise 8.0 Preliminary Upgrade Steps

Perform the procedures in the following sections before upgrading Access Manager or Federation Manager to OpenSSO Enterprise 8.0.

Upgrading Related Components

The components on the host machine of the previous release's installation must also be supported by Open SSO Enterprise 8.0. If necessary, upgrade the following components, in this order:

- Operating system
- Sun Java System Directory Server
- Access Manager web container
- JDK (must be 1.5 or later)

For a list of the supported versions of these components, see "Hardware and Software Requirements For OpenSSO Enterprise 8.0" in *Sun OpenSSO Enterprise 8.0 Release Notes*.

Note – If upgrading multiple instances of Access Manager or Federation Manager, upgrade these components on all host machines.

Backing Up Existing Access Manager or Federation Manager Files

Before upgrading Access Manager 7.1, Access Manager 7 2005Q4, or Federation Manager 7.0, back up the files as described in the following sections.

Note – If upgrading multiple instances of Access Manager or Federation Manager, backup these files for all instances.

To Back Up the Access Manager or Federation Manager Schema

Back up the Access Manager or Federation Manager schema (and any corresponding attribute values) that was loaded to Sun Java System Directory Server during installation. Export the data to an LDIF file using the appropriate Directory Server command.

- If backing up Directory Server 6.x, use the `dsadm export` command. For more information, see the `dsadm(1M)` man page.
- If backing up Directory Server 5.x, use the `db2ldif` command. For more information, see the `db2ldif(1M)` man page.

You will need to know the base suffix of the service management node (also referred to as the *information tree*) as defined in the `amsamplesilent` file. For information on the suffix of this node, see "Configuring Different Root Suffixes for the Access Manager Information Tree and User Directory Nodes" in *Sun Java System Access Manager 7.1 Postinstallation Guide*.

Caution – Because OpenSSO Enterprise does not require the `iPlanetAMProviderConfigService` and `iPlanetAMAuthenticationDomainConfigService`, the upgrade process removes these services from the schema. If you do not back up the schema, retrieval of these services is not possible after the upgrade is finished.

To Back Up the Access Manager or Federation Manager Configuration Data

Back up the Access Manager or Federation Manager configuration data stored in the `AMConfig.properties` and `severconfig.xml` files. The location of the files is based on your platform.

- Solaris systems: `/etc/opt/SUNWam/config`
- Linux systems: `/etc/opt/sun/identity/config`
- Windows systems (for Access Manager 7.1 only): `C:\Program Files\Sun\JavaES5\identity\config`

To Back Up Customized Files

Back up any files that might have been customized for your Access Manager or Federation Manager deployment, including:

- JSP files customized for the Access Manager console
- JAR files for authentication and customized modules in the *AccessManager-base*`/lib` directory, where *AccessManager-base* represents the base installation directory based on your platform.
  - Solaris systems: `/opt/SUNWam`
  - Linux systems: `/opt/sun/identity`

- Windows systems (for Access Manager 7.1 only): `C:\Program Files\Sun\JavaES5\identity`

To Back Up the Federation Manager 7.0 Staging Directory

If you are upgrading Federation Manager 7.0, back up the Federation Manager staging directory. For information about the staging directory see the *Sun Java System Federation Manager 7.0 User's Guide*.

To Modify Policy Definitions

Modify any policy definitions that meet the criteria as documented in "Modifying Policy Definitions" on page 24. This can be done before or after the upgrade.

Setting Your `JAVA_HOME` and `PATH` Environment Variables

The upgrade scripts and `jar` command require JDK 1.5 or later. Therefore, set your `JAVA_HOME` environment variable to point to an installed JDK version 1.5 or later. Additionally, set your PATH environment variable. For example, run the following commands if using Solaris.

```
JAVA_HOME=/usr/java/j2sdk1.5.0
export JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH;
export PATH
```

Confirm the changes made by running `which java` at the command line. Run `java -version` to confirm the version number.

Collecting Data Required for Upgrade

During the upgrade process, you will need to know the data described in the following sections. Most of this information is stored in the `AMConfig.properties` and `severconfig.xml` files.

Access Manager or Federation Manager Settings

Collect the following information regarding the previously installed server instances.

- Administrator (by default, `amadmin`) password
- Host name

- Port
- Cookie domain
- Platform locale (defined by `com.iplanet.am.locale`)
- Default Policy Agent user (`UrlAccessAgent`) password (usually the same as the `amldapuser` password)
- Deployment URI of the existing Access Manager or Federation Manager instance (by default, `amserver` or `federation`, respectively)

Directory Server Settings for the Configuration Data Store

Collect the following information regarding the Directory Server configuration data store.

- Disable SSL before you begin the upgrade process. See the `com.iplanet.am.directory.ssl.enabled` property in `AMConfig.properties` from the previously installed release.
- Host name
- Port
- Encryption key: Use the value of the `am.encryption.pwd` property from `AMConfig.properties` from the previously installed release.
- Root suffix: Use the value of the `com.iplanet.am.rootsuffix` property in `AMConfig.properties` from the previously installed release.
- Administrator (by default, `cn=Directory Manager`)
- Administrator password
- `amldapuser` password

Directory Server Settings for the User Data Store

Collect the following information regarding the Directory Server user data store.

- Disable SSL before you begin the upgrade process. See the `com.iplanet.am.directory.ssl.enabled` property in `AMConfig.properties` from the previously installed release.
- Host name
- Port
- Root suffix: Use the value of the `com.iplanet.am.rootsuffix` property in `AMConfig.properties` from the previously installed release.
- Administrator (by default, `cn=Directory Manager`)

- Administrator password

Upgrading to OpenSSO Enterprise 8.0

Use the following list of procedures to upgrade the previous release to OpenSSO Enterprise 8.0.

1. "Getting OpenSSO Enterprise 8.0" on page 11
2. "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12
3. "Removing the Previously Installed Server Software" on page 13
4. "Deploying the Open SSO Enterprise 8.0 WAR" on page 14
5. "Running the Pre-Upgrade Script" on page 15
6. "Modifying `ssoUpgradeConfig.properties` and `AMConfig.properties.bak`" on page 17
7. "Configuring Open SSO Enterprise 8.0 Against the Existing Access Manager or Federation Manager Schema" on page 19
8. "Upgrading the Existing Access Manager or Federation Manager Schema" on page 21

Getting OpenSSO Enterprise 8.0

OpenSSO Enterprise 8.0 is distributed as a downloadable ZIP file named `opensso_enterprise_80.zip`. This ZIP file contains functionality originally developed for both Access Manager and Federation Manager functionality, as well as new OpenSSO Enterprise 8.0 features.

▼ To Download and Unzip `opensso_enterprise_80.zip`

1 **Log in to the host machine as super user (`root`).**

2 **Create a base directory in which to download and unzip `opensso_enterprise_80.zip`.**

This guide uses `zip-root` as the name of the base directory. You must have both read and write access to this directory.

```
# mkdir /zip-root
# cd /zip-root
```

3 **Download the `opensso_enterprise_80.zip` file from the following site to the `zip-root` directory.**

Sun Downloads site under View by Category, Identity Management, and then OpenSSO Enterprise: `http://www.sun.com/download/index.jsp`

4 **Unzip `opensso_enterprise_80.zip`.**

The upgrade scripts and related files are in the `zip-root/opensso/upgrade` directory.

Note – Check the permissions on the `ssopre80upgrade` and `ssoupgrade` scripts. If these scripts do not have the execute permission, reset the permissions using `chmod`.

Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR

After unzipping `opensso_enterprise_80.zip`, you will find `opensso.war` in `/zip-root/opensso/deployable-war`. Now create a staging directory into which `opensso.war` can be exploded. If you customized any files in your previous Access Manager or Federation Manager deployment, you will need to apply these customizations to the files in the staging directory.

Note – Even if you have not customized your previous deployment, you must create a staging directory as the path to this directory is used for input when running the upgrade scripts.

▼ To Create a Staging Directory and OpenSSO Enterprise 8.0 WAR

Before You Begin This procedure assumes you are still logged in to the host machine as super user.

1 **Create a staging directory for the new WAR.**

Caution – Do not create the staging directory in the `tmp` directory.

```
# mkdir /staging
```

2 **Extract the files from `opensso.war` into the staging directory.**

```
# cd /staging
# jar xvf /zip-root/opensso/deployable-war/opensso.war
```

3 **(Optional) Add any customized files from your previous Access Manager or Federation Manager deployment to the OpenSSO Enterprise WAR staging directory.**

For example, copy any modified JSP files for the Administration Console to the staging directory.

Note – Be sure to remove the file you are replacing.

4 **Create a new WAR from the files in the staging directory.**

The name of the new WAR must use the same deployment URI as the previously installed Access Manager or Federation Manager instance. For example, if the previous instance is deployed with the `/amserver` URI, the new WAR must be named `amserver.war`.

```
# jar cvf /zip-root/opensso/deployable-war/amserver.war *
```

Removing the Previously Installed Server Software

Remove previously installed software dependant on what was installed and how it was initially installed. You might have to do one or all of the following depending on your deployment.

- For an Access Manager 7.1 or a Federation Manager WAR deployment, undeploy the appropriate WAR (`amserver.war` for Access Manager or `federation.war` for Federation Manager) using the web container console or command line interface. For information, see the documentation for the web container.
- For a Java Enterprise System installer deployment of Access Manager 7.1 or Access Manager 7 2005Q4, undeploy all web applications (`amcommon`, `amconsole`, `ampassword` and `amserver`) by executing the `amconfig` script with `DEPLOY_LEVEL=26` as input in the `amsamplesilent` file. For more information, see Chapter 2, "Running the Access Manager amconfig Script," in *Sun Java System Access Manager 7.1 Postinstallation Guide*.
- After you undeploy the appropriate WAR file or undeploy all web applications, remove the `classpath` for the old Access Manager or Federation Manager libraries.

  For example, if using Application Server 9.1, edit `/opt/SUNWappserver/appserver/domains/domain1/config/domain.xml` to remove all paths containing `/opt/SUNWam/`. Remove all entries containing `SUNWam` and `SUNWma`.

  - ✓ `/opt/SUNWam/lib/xmlsec.jar`
  - ✓ `/etc/opt/SUNWam/config`
  - ✓ `/opt/SUNWam/lib`
  - ✓ `/opt/SUNWam/locale`
  - ✓ `/opt/SUNWam/lib/am_sdk.jar`
  - ✓ `/opt/SUNWam/lib/ldapjdk.jar`
  - ✓ `/opt/SUNWam/lib/am_services.jar`
  - ✓ `/opt/SUNWam/lib/am_sso_provider.jar`
  - ✓ `/opt/SUNWam/lib/swec.jar`
  - ✓ `/opt/SUNWam/lib/acmecrypt.jar`
  - ✓ `/opt/SUNWam/lib/iaik_ssl.jar`
  - ✓ `/opt/SUNWam/lib/iaik_jce_full.jar`
  - ✓ `/opt/SUNWam/lib/am_logging.jar`
  - ✓ `/opt/SUNWam/lib/jce1_2_1.jar`
  - ✓ `/opt/SUNWma/lib/wireless_rendering.jar`
  - ✓ `/opt/SUNWma/lib/wireless_rendering_util.jar`
  - ✓ `/opt/SUNWma/lib/mobile_services.jar`

- ✓ /opt/SUNWma/lib/ccpp-1_0.jar
- ✓ /opt/SUNWma/lib/ccpp-ri-1_0.jar
- ✓ /opt/SUNWma/lib/jena-1.4.0.jar
- ✓ /opt/SUNWma/lib/rdffilter.jar
- ✓ /opt/SUNWma/lib/locale
- ✓ /opt/SUNWam/lib/mobile_identity.jar

When deployed on a Linux system, the entries are:

- ✓ /etc/opt/sun/identity/config
- ✓ /opt/sun/identity/lib
- ✓ /opt/sun/identity/locale
- ✓ /opt/sun/identity/lib/am_sdk.jar
- ✓ /opt/sun/share/lib/ldapjdk.jar
- ✓ /opt/sun/identity/lib/am_services.jar
- ✓ /opt/sun/identity/lib/am_sso_provider.jar
- ✓ /opt/sun/identity/lib/swec.jar
- ✓ /opt/sun/identity/lib/acmecrypt.jar
- ✓ /opt/sun/identity/lib/iaik_ssl.jar
- ✓ /opt/sun/identity/lib/iaik_jce_full.jar
- ✓ /opt/sun/identity/lib/am_logging.jar
- ✓ /opt/sun/mobileaccess/share/lib/wireless_rendering.jar
- ✓ /opt/sun/mobileaccess/share/lib/wireless_rendering_util.jar
- ✓ /opt/sun/mobileaccess/share/lib/mobile_services.jar
- ✓ /opt/sun/mobileaccess/share/lib/ccpp-1_0.jar
- ✓ /opt/sun/mobileaccess/share/lib/ccpp-ri-1_0.jar
- ✓ /opt/sun/mobileaccess/share/lib/jena-1.4.0.jar
- ✓ /opt/sun/mobileaccess/share/lib/rdffilter.jar
- ✓ /opt/sun/mobileaccess/share/lib/locale
- ✓ /opt/sun/identity/lib/mobile_identity.jar

Note – See OpenSSO issue 4032 for more information.

Deploying the Open SSO Enterprise 8.0 WAR

In this procedure, deploy the new OpenSSO Enterprise 8.0 WAR created in "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12.

▼ To Deploy the Open SSO Enterprise 8.0 WAR

Before You Begin

- This procedure assumes you are still logged in to the host machine as super user.

- Be sure to follow the container-specific instructions in Chapter 2, "Deploying the OpenSSO Enterprise Web Container," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide* before deploying the WAR — in particular, read "Adding Security Permissions For a Web Container" in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

1 **Deploy the OpenSSO Enterprise WAR using your web container's administration console or command line interface.**

Use the same host name, port and URI (`/amserver`) on which the previous Access Manager or Federation Manager instance was deployed. In this example, `amserver.war` (created in "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12) is located in `/zip-root/opensso/deployable-war`.

2 **Restart the OpenSSO Enterprise web container.**

Running the Pre-Upgrade Script

Run the `ssopre80upgrade` script (`ssopre80upgrade.bat` on Windows) when you are upgrading package based installations; for example, instances of Access Manager installed using Java Enterprise System (JES) 4 or JES 5 will put Access Manager packages on the host machine but a WAR that has been downloaded and deployed will not. `ssopre80upgrade` prepares the system for upgrade by:

- Backing up existing Access Manager or Federation Manager files (such as logs and configuration files).
- Removing the Access Manager 7.1 or Access Manager 7 2005Q4 packages (except on Windows systems).
- Removing the Federation Manager 7.0 packages.
- Removing the SAMLv2 Plug-in package.
- Updating the `/var/sadm/install/productregistry` file to reflect the removal of the Java Enterprise System Access Manager packages.

Note – Entering path names on Windows When you run the `ssopre80upgrade.bat` script, replace each backslash (`\`) in path names to a forward slash (`/`). For example, for `C:\sun\opensso\config`, you would enter `C:/sun/opensso/config`.

▼ To Execute the `ssopre80upgrade` Script

Before You Begin

- This procedure assumes you are still logged in to the host machine as super user.
- Copy (and rename by adding the `.bak` extension) the previous version's configuration files (`AMConfig.properties` and `serverconfig.xml`) to the `/zip-root/opensso/upgrade/config` directory.

```
# cp AMConfig.properties AMConfig.properties.bak
# mv AMConfig.properties.bak /zip-root/opensso/upgrade/config/
```

This step is needed for the next procedure, "Modifying ssoUpgradeConfig.properties and AMConfig.properties.bak" on page 17.

- **If upgrading a Linux-based installation,** copy the `ssoupgrade` file to a new directory (for example `/tmp/ssoupgrade`). After executing `ssopre80upgrade` add values for the following attributes in the `ssoupgrade` file and return the modified script to the `/zip-root/opensso/upgrade/scripts` directory for execution in "Upgrading the Existing Access Manager or Federation Manager Schema" on page 21.
  - `LIB_DIR=`*STAGING_DIR*: path to the directory created in "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12. For example: `LIB_DIR=/staging`
  - `CONFIG_DIR=`*OSSO_CONFIG_DIR*: path to the configuration directory for the newly upgraded instance. For example: `CONFIG_DIR=/amserver`
  - `UPGRADE_DIR=`*OSSO_UPGRADE_DIR*: path to the upgrade directory in the exploded `opensso.zip`. For example: `UPGRADE_DIR=/zip-root/opensso/upgrade`

1. **Change to the `/zip-root/opensso/upgrade/scripts` directory.**

2. **Run the `ssopre80upgrade` script.**

   On Solaris and Linux systems, enter `./ssopre80upgrade`. On Windows systems, enter `ssopre80upgrade.bat`. The script checks whether you are upgrading an instance of Access Manager or Federation Manager.

   ```
   # check_instance isFM is : false
   # A log of the ssopreupgrade process can be found in
   /var/sadm/install/logs/Sun_Java_System_Access_Manager_
   upgrade_log.03101744
   ```

3. **When prompted, accept the default values or provide information based on the deployment.**
   - Directory Server fully qualified host name
   - Directory Server port
   - Top-level administrator DN (`uid=amAdmin,ou=People,dc=sun,dc=com`)
   - Top-level administrator (`uid=amAdmin,ou=People,dc=sun,dc=com`) password
   - Directory to store backup files (`/opt/SUNWam`)
   - OpenSSO Enterprise 8.0 configuration directory is the path to the configuration directory for the *original* instance of Access Manager that you are upgrading. For example: `/etc/opt/SUNWam`
   - OpenSSO Enterprise 8.0 upgrade directory is the path to the exploded OpenSSO ZIP. For example: `/zip-root/opensso`

- OpenSSO Enterprise 8.0 staging directory is the path to the directory in which the OpenSSO WAR was exploded. For example: `/staging`
- **OPTIONAL:** Access Manager installation directory (Windows only)
- **OPTIONAL:** Federation Manager 7.0 staging directory (Federation Manager only)

After the final value is entered, `ssopre80upgrade` removes the previous version of the server software and its packages.

Next Steps **If upgrading a Linux installation** remember to edit `ssoupgrade` and return it to the `/zip-root/opensso/upgrade/scripts` directory.

Modifying `ssoUpgradeConfig.properties` and `AMConfig.properties.bak`

In preparation for the final procedures in the upgrade process, modify two properties files.

- `ssoUpgradeConfig.properties` is used by the `ssoupgrade` script.
- `AMConfig.properties.bak` defines an encryption method not supported by OpenSSO Enterprise 8.0.

▼ To Modify `ssoUpgradeConfig.properties` and `AMConfig.properties.bak`

Before You Begin This procedure assumes you are still logged in to the host machine as super user.

1. **Rename the previous version's configuration files (`AMConfig.properties` and `serverconfig.xml`) using the `.bak` extension.**

```
# cp AMConfig.properties AMConfig.properties.bak
# cp serverconfig.xml serverconfig.xml.bak
```

2. **Move the renamed configuration files (`AMConfig.properties.bak` and `serverconfig.xml.bak`) to the `/zip-root/opensso/upgrade/config` directory.**

```
# mv AMConfig.properties.bak /zip-root/opensso/upgrade/config/
# mv serverconfig.xml.bak /zip-root/opensso/upgrade/config/
```

3. **Change to the `/zip-root/opensso/upgrade/config/` directory.**

4. **Open `ssoUpgradeConfig.properties` in a text editor and set the following properties.**

   - `XML_ENCODING`: For example: `XML_ENCODING=UTF-8`
   - `BASEDIR`: The directory in which the OpenSSO Enterprise 8.0 ZIP is exploded. For example: `BASEDIR=/zip-root/opensso`

- `ORG_NAMING_ATTR`: Organization naming attribute. Default is `o`. For example: `ORG_NAMING_ATTR=o`
- `USER_NAMING_ATTR`: User naming attribute. Default is `uid`. For example: `USER_NAMING_ATTR=uid`
- `DEPLOY_URI`: OpenSSO Enterprise deployment URI. For example: `DEPLOY_URI=amserver`
- `PAM_SERVICE_NAME`:
  - Solaris systems: `PAM_SERVICE_NAME=other`
  - Linux systems: `PAM_SERVICE_NAME=password`

Note – PAM is the Pluggable Authentication Module for Unix. The defined name is based on the operating system flavor.

- `DB_NAME`: OpenSSO Enterprise back-end database. Default name: `userRoot`
- `INSTANCE_TYPE`: Set to the instance type you are upgrading:
  - Access Manager: `INSTANCE_TYPE=AM`
  - Federation Manager: `INSTANCE_TYPE=FM`
- `LDAP_USER_PASS`: `amldapuser` password
- `ORG_OBJECT_CLASS=sunismanagedorganization` is the default.
- `USER_OBJECT_CLASS=inetorgperson` is the default.

5 **Save** `ssoUpgradeConfig.properties`**.**

6 **Open** `AMConfig.properties.bak` **in a text editor and make the following modifications.**

Earlier versions of OpenSSO Enterprise supported, by default, Java Security Services (JSS) encryption. Now, by default, OpenSSO Enterprise supports Java Cryptography Extension (JCE) encryption.

a. **Change the value of** `com.iplanet.security.SecureRandomFactoryImpl` **from** `com.iplanet.am.util.JSSSecureRandomFactoryImpl` **to** `com.iplanet.am.util.SecureRandomFactoryImpl`**.**

b. **Change the value of** `com.iplanet.security.SSLSocketFactoryImpl` **from** `com.iplanet.services.ldap.JSSSocketFactory` **to** `netscape.ldap.factory.JSSESocketFactory`**.**

c. **Change the value of** `com.iplanet.security.encryptor` **from** `com.iplanet.services.util.JSSEncryption` **to** `com.iplanet.services.util.JCEEncryption`**.**

Caution – If you use a web container not developed and branded by Sun Microsystems (something other than Glassfish, Web Server and Application Server), it is recommended to make this change.

7 **Save** `AMConfig.properties.bak`.

Configuring Open SSO Enterprise 8.0 Against the Existing Access Manager or Federation Manager Schema

After you deploy the OpenSSO Enterprise WAR, configure the new OpenSSO Enterprise instance against the existing Access Manager or Federation Manager schema using the Configurator.

Note – You can also use command-line configuration as described in Chapter 5, "Configuring OpenSSO Enterprise Using the Command-Line Configurator," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

▼ To Configure OpenSSO Enterprise Against the Existing Access Manager or Federation Manager Schema

Before You Begin This procedure assumes you have deployed the OpenSSO Enterprise WAR.

1 **Launch the GUI Configurator by entering the OpenSSO Enterprise URL in your browser.**

Use the format *protocol*`://`*server-host*`:`*server-port*`/`*deployuri*. For example: `http://abc.example.com:8080/amserver`.

2 **On the Configuration Options page, click Create New Configuration.**

3 **On the Default User Password page, enter and confirm the `amAdmin` password.**

Use the same password as the one defined for the Access Manager or Federation Manager instance you are upgrading.

4 **Click Next to continue.**

5 **On the Server Settings page, enter values for the following:**

- **Server URL**: Use the same value as the one defined for the Access Manager or Federation Manager instance you are upgrading

- **Cookie Domain**: Use the same value as the one defined for the Access Manager or Federation Manager instance you are upgrading
- **Platform Locale**: Use the same value as the one defined for the Access Manager or Federation Manager instance you are upgrading
- **Configuration Directory**: Use the default value (`/opensso`) or specify another value.

6 Click Next to continue.

7 On the Configuration Data Store Settings, do the following:

a. Check First Instance.

b. For Configuration Data Store, check Sun Java System Directory Server.

c. Specify the following Directory Server values from the existing Access Manager or Federation Manager instance:

- **SSL Enabled**: Disable for the upgrade process.
- **Host Name**
- **Port**
- **Encryption Key**
- **Root Suffix**: Enter the root suffix defined by the installation of the older version being replaced.
- **Login ID**: Directory Server Administrator DN
- **Password**: Directory Server Administrator password

8 Click Next to continue.

9 On the User Data Store Settings page, do the following:

Click **Use Other User Data Store** to specify Sun Java System Directory Server and specify the Directory Server values from the existing Access Manager or Federation Manager instance:

- **SSL Enabled**: Disable for the upgrade process.
- **Directory Name**: In this example, it is the same as the Host Name without the domain qualifiers.
- **Port**:
- **Root Suffix**: Use a different root from the one defined for the Configuration Data Store.
- **Login ID**: Directory Server Administrator DN
- **Password**: Directory Server Administrator password
- **User Data Store Type**: Check LDAP with OpenSSO Schema

10 **Click Next to continue.**

11 **On the Site Configuration page, check No and click Next to continue.**

12 **On the Default Policy Agent User page, enter and confirm the password for the Policy Agent user.**

The default Policy Agent user is `UrlAccessAgent`. The password is usually the same as the password of `amldapuser`.

13 **Click Next to continue.**

14 **Verify that the Configuration Summary Details are correct and click Create Configuration.**

When the configuration is complete, the Configurator displays a link to redirect you to the OpenSSO Enterprise administration console.

15 **Log in to the OpenSSO Enterprise Administration Console as `amadmin` using the Data Store authentication module and the password specified during configuration.**

The URL to access the Data Store authentication module is formatted as **`http://`***host*`:`*port*`/`*deployURI*`/UI/Login?module=DataStore`. At this point, OpenSSO Enterprise is running against the existing Access Manager or Federation Manager schema (or DIT), which is known as coexistence mode.

Tip – If upgrading from Legacy mode to Realm mode, a login at this point will fail. You should skip this step and login after completing "Upgrading the Existing Access Manager or Federation Manager Schema" on page 21.

Upgrading the Existing Access Manager or Federation Manager Schema

The `ssoupgrade` (or `ssoupgrade.bat` on Windows) script upgrades the Access Manager or Federation Manager schema (previously used to configure the deployed OpenSSO Enterprise WAR) to the OpenSSO Enterprise 8.0 schema.

Note – Entering path names on Windows. When you run `ssoupgrade.bat` on Windows, replace each backslash (`\`) in path names to a forward slash (`/`). For example, you would enter `C:/sun/opensso/config` for `C:\sun\opensso\config`.

▼ To Upgrade the Access Manager or Federation Manager Schema Using the `ssoupgrade` Script

Before You Begin Before executing `ssoupgrade`, rename (by adding the `.bak` extension) and then copy the previous version's configuration files (`AMConfig.properties` and `serverconfig.xml`) to the `/zip-root/opensso/upgrade/config` directory.

```
# cp AMConfig.properties AMConfig.properties.bak
# mv AMConfig.properties.bak /zip-root/opensso/upgrade/config/
```

1. **Log on as super user (`root`).**

2. **Change to the `zip-root/opensso/upgrade/scripts` directory.**

3. **(Optional) Change the values for the following properties in the `ssoupgrade` script.**

   If you don't run `ssopre80upgrade` these properties would not be updated to reflect the new configuration. This is most common in WAR based deployments for which you do not need to run `ssopre80upgrade`. Thus when upgrading WAR based deployments, `ssoupgrade` needs to be manually edited before executing it.

   - `LIB_DIR=`*STAGING_DIR*: path to the directory created in "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12. For example: `LIB_DIR=/staging`
   - `CONFIG_DIR=`*OSSO_CONFIG_DIR*: path to the configuration directory for the newly upgraded instance. For example: `CONFIG_DIR=/amserver`
   - `UPGRADE_DIR=`*OSSO_UPGRADE_DIR*: path to the upgrade directory in the exploded `opensso.zip`. For example: `UPGRADE_DIR=/zip-root/opensso/upgrade`

   Tip – The `ssoupgrade` script might need modification even if the `ssopre80upgrade` script was executed. Confirm that values for these properties have been swapped correctly before executing the script.

4. **Run the `ssoupgrade` script.**

   - Solaris and Linux systems: `./ssoupgrade`
   - Windows: `ssoupgrade.bat`

5. **When prompted, provide the following information.**

   - Upgrade Base Directory: path to the directory created after exploding the `opensso.zip`. For example: `/zip-root/opensso`
   - OpenSSO Configuration Directory: path to the directory created after deploying the `opensso.war`. For example: `/opensso`
   - OpenSSO Staging Directory: path to the directory created in "Creating a Staging Directory and OpenSSO Enterprise 8.0 Upgrade WAR" on page 12. For example: `/staging`

- Fully qualified host name of the machine on which the Directory Server is installed
- Directory Server port number. For example: `389`
- Directory Manager DN. For example, `cn=Directory Manager`
- Directory Manager password
- OpenSSO Administrative User DN: the DN of `amAdmin`. By default, `uid=amAdmin,ou=People,dc=sun,dc=com`
- OpenSSO Administrative User password (`amAdmin` password)
- Enable Realms

  This prompt is displayed only if the existing instance is in Legacy mode or is an instance of Federation Manager. To migrate to Realm mode, enter **`y`**. Sun recommends that you migrate to Realm mode.

After entering the script values, the process begins.

6 **Restart the Open SSO Enterprise web container.**

Next Steps Log in to the OpenSSO Enterprise Administration Console as `amadmin` using the Data Store authentication module and the password specified during configuration. The URL to access the Data Store authentication module is:

`http://`*`host`***`:`***`port`***`/`***`deployURI`***`/UI/Login?module=DataStore`**

Post-Upgrade Tasks

The following sections contain post-migration tasks for some specific deployment issues.

Migrating Roles from the Old LDAPv3 Plugin to the OpenSSO Enterprise Sun DS Plugin

OpenSSO Enterprise 8.0 does not support role management or password management when using the Generic LDAPv3 data store plugin. If the instance you are upgrading is configured to use this plugin, follow the instructions in Chapter 15, "Enabling the Access Manager SDK (AMSDK) Identity Repository Plug-in," in *Sun OpenSSO Enterprise 8.0 Installation and*

Configuration Guide to enable the AMSDK Identity Repository plugin. Alternately, you can add a new Sun DS data store using the OpenSSO Enterprise schema, point to the same LDAPv3 directory server, and remove the LDAPv3 data store plugin when this has been finished.

Configuring OpenSSO Enterprise for the Apache Agent

The Policy Agent 2.2 for Apache encodes the `appssotoken` cookies, but OpenSSO Enterprise 8.0 does not decode them properly. To decode properly, enable cookie encoding on the server side using the following procedure.

▼ To Enable Cookie Encoding for the Apache Agent

1. **Log in to the OpenSSO Enterprise console as administrator; by default, `amadmin`.**
2. **Click the Configuration tab.**
3. **Under Servers and Sites, click Default Server Settings.**
4. **Click the Security tab.**
5. **Under Cookie, enable Encode Cookie Value.**

   Be sure to enable this attribute on each individual server either individually or through inheritance.
6. **Click Save.**
7. **Log out of the OpenSSO Enterprise console.**

Modifying Policy Definitions

With the release of OpenSSO Enterprise 8.0, policy evaluation for URL pattern matching of rules with query parameters no longer match the generic asterisk (*); you must explicitly allow query parameters for the URL policies. For those URLs which include query parameters, the policy definition must include the following rules.

- **`http*://`***host*`:`*port*`/appcontext/*`**
- **`http*://`***host*`:`*port*`/appcontext/*?*`**

Note – This modification can be done before the upgrade as Access Manager 7.x will evaluate these additional rules without issue.

Uninstall Packages on Windows

On Windows, you must the uninstall the Access Manager packages manually. For information, see the *Sun Java Enterprise System 5 Installation Guide for Microsoft Windows*.

Remove Federation Manager Staging Directory

You can manually remove the Federation Manager 7.0 staging directory.

Upgrading Multiple Instances of Access Manager

The following sections describe procedures when upgrading multiple instances of Access Manager.

Preliminary Steps For Upgrading Multiple Instances of Access Manager

Follow the instructions in "OpenSSO Enterprise 8.0 Preliminary Upgrade Steps" on page 7 and "Collecting Data Required for Upgrade" on page 9 to complete the preliminary steps.

Upgrading Multiple Instances of Access Manager 7.1

Upgrading multiple instances of Access Manager 7.1 is based on coexistence: an instance of OpenSSO Enterprise 8.0 can coexist with instances of Access Manager 7.1 in the same deployment and can concurrently access the Access Manager 7.1 schema in Directory Server. Thus, upgrade the instances one at a time. Upgrading multiple instances of Access Manager 7.1 is supported when:

- Two or more Access Manager 7.1 instances are deployed in supported web containers on different hosts behind a load balancer. The instances were deployed either by running the Sun Java Enterprise System installer (package based installation) or from a WAR file.

Note – If Access Manager 7.1 was deployed from a WAR file, the configuration data store must be in Sun Java System Directory Server and not in the Flat File System.

- The configuration data store is in Sun Java System Directory Server; the Directory Server is set up for multi-master replication.

Note – If multiple Access Manager 7.1 instances point to a configuration data store in a single Directory Server instance, first upgrade all Access Manager 7.1 instances sequentially and then upgrade the Directory Server schema.

▼ To Upgrade Multiple Access Manager 7.1 Server Instances With Directory Server Configured For MMR

Before You Begin Follow the instructions in "OpenSSO Enterprise 8.0 Preliminary Upgrade Steps" on page 7 and "Collecting Data Required for Upgrade" on page 9 to complete the preliminary steps.

1. **Back up the Access Manager 7.1 files as described in "Backing Up Existing Access Manager or Federation Manager Files" on page 7.**
2. **Upgrade the first Access Manager 7.1 instance using the following sub procedure.**
   a. **Disable the Access Manager 7.1 instance in the load balancer.**
      Requests will no longer be routed to this instance.
   b. **Upgrade the first Access Manager instance as described in "Upgrading to OpenSSO Enterprise 8.0" on page 11.**
   c. **Enable the upgraded OpenSSO Enterprise 8.0 instance in the load balancer.**
      Requests will once again be routed to this instance.
3. **Follow the same procedure for all other Access Manager 7.1 instances in the deployment sequentially:**
4. **Upgrade the Directory Server schema as described in "Upgrading the Existing Access Manager or Federation Manager Schema" on page 21.**
   Requests will now be routed through the load balancer to all upgraded OpenSSO Enterprise 8.0 instances in the deployment using the upgraded schema in Directory Server.

Upgrading Multiple Access Manager 7 2005Q4 Server Instances

Instances of OpenSSO Enterprise 8.0 cannot coexist in the same deployment with the Access Manager 7 2005Q4 Directory Server schema. However, if Directory Server is set up for multi-master replication, you can upgrade. Upgrading multiple instances of these earlier versions of Access Manager is supported when:

- Two or more Access Manager 7 2005Q4 instances are deployed in supported web containers behind a load balancer. The instances were deployed by running the Sun Java Enterprise System installer (package based installation).
- The configuration data store is in Sun Java System Directory Server, which is set up for multi-master replication.

Note – If multiple Access Manager 7 2005Q4 instances point to a configuration data store in a single Directory Server instance, first upgrade all Access Manager instances sequentially and then upgrade the Directory Server schema.

Caution – If the Directory Server is not configured for multi-master replication, you cannot perform a rolling upgrade. Therefore, there will be downtime while upgrading the Directory Server schema.

▼ To Upgrade Multiple Instances of Access Manager 7 2005Q4 With Directory Server Configured For Multi-Master Replication

This procedure assumes two instances of Directory Server.

Before You Begin Follow the instructions in "OpenSSO Enterprise 8.0 Preliminary Upgrade Steps" on page 7 and "Collecting Data Required for Upgrade" on page 9 to complete the preliminary steps.

1. **Modify the configuration for the first Access Manager instance so that it points to the second Directory Server instance rather than the first Directory Server instance.**

2. **Restart the first Access Manager instance.**

   The first Access Manager instance will continue handling requests while you upgrade the other Access Manager instances in the deployment.

3. **Upgrade all other instances of Access Manager sequentially using the following sub procedure.**

   a. **Disable the Access Manager instance in the load balancer.**

      Requests will no longer be routed to this instance.

b. **Upgrade the Access Manager instance as described in "Upgrading to OpenSSO Enterprise 8.0" on page 11.**

c. **Disable Directory Server MMR in the first instance of Directory Server.**

d. **Update the schema for the first Directory Server instance as described in "Upgrading the Existing Access Manager or Federation Manager Schema" on page 21.**

e. **Restart the upgraded OpenSSO Enterprise 8.0 instance.**

f. **Enable the upgraded OpenSSO Enterprise 8.0 instance in the load balancer.**

Requests once again will be routed to this instance.

4 **Upgrade the first Access Manager instance.**

a. **Disable the first Access Manager instance in the load balancer.**

Requests will no longer be routed to this instance.

b. **Upgrade the first Access Manager instance as described in "Upgrading to OpenSSO Enterprise 8.0" on page 11.**

c. **Enable Directory Server MMR in the first instance of Directory Server.**

The schema for the second Directory Server instance is now updated to the OpenSSO Enterprise 8.0 schema (as well as any other Directory Server instances).

d. **Restore the configuration of the first upgraded OpenSSO Enterprise 8.0 instance to point to the first Directory Server instance.**

e. **Restart the first upgraded OpenSSO Enterprise 8.0 instance.**

f. **Enable the first upgraded OpenSSO Enterprise 8.0 instance in the load balancer.**

Requests will be routed through the load balancer to all upgraded OpenSSO Enterprise 8.0 instances in the deployment, using the upgraded schema in Directory Server.

Implementing Session Failover After Upgrades

Information on implementing session failover after upgrading multiple instances to OpenSSO Enterprise 8.0 can be found in Chapter 8, "Implementing OpenSSO Enterprise Session Failover," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

Using Policy Agents After Upgrading to OpenSSO Enterprise

See also "Configuring OpenSSO Enterprise for the Apache Agent" on page 24

Migrating a Version 2.2 Policy Agent to Version 3.0

In this scenario, you have upgraded an Access Manager 7.1 or Access Manager 7 2005Q4 deployment to OpenSSO Enterprise 8.0 and you also want to migrate an existing version 2.2 policy agent to version 3.0. After you migrate a version 2.2 agent, the agent can use the new version 3.0 agent features.

Note – Before you can migrate a version 2.2 agent, a corresponding version 3.0 agent must exist. Some version 3.0 agents are available as patch releases. To determine the available version 3.0 agents, check the agent guides in the following documentation collection: `http://docs.sun.com/coll/1767.1`.

To migrate a version 2.2 policy agent to version 3.0, the version 3.0 `agentadmin` program includes the new `--migrate` option. The `--migrate` option performs these functions for a 2.2 agent:

- Migrates the agent's binary files
- Updates the agent's container configuration
- Converts the agent's `AMAgent.properties` file to the new version 3.0 `OpenSSOAgentBootstrap.properties` and `OpenSSOAgentConfiguration.properties` files
- Creates new deployment directories for migrated 3.0 agents, starting with `Agent_001`. The program does not create a one-to-one mapping of directories. For example, if the 2.2 agents have the `Agent_001` and `Agent_003` directories (`Agent_002` was removed), the migrated 3.0 agents will have the `Agent_001` and `Agent_002` directories.

The `agentadmin` program does not modify the version 2.2 agent deployment directory files in case you need these files after you migrate.

Migrating a version 2.2 agent involves these general steps:

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

   To get the version 3.0 `agentadmin` program, you must download the version 3.0 agent that corresponds to the version 2.2 agent you are migrating. For example, if you are migrating the version 2.2 Apache HTTP Server agent, download the corresponding version 3.0 Apache HTTP Server agent.

2. On the OpenSSO Enterprise server, run the `ssoadm` utility to create the new version 3.0 agent configuration in the OpenSSO Enterprise centralized agent configuration repository.

   You must use the `ssoadm` utility from the `openssoAdminTools.zip` file on the OpenSSO Enterprise server. For information, see Chapter 6, "Installing the OpenSSO Enterprise Utilities and Scripts," in *Sun OpenSSO Enterprise 8.0 Installation and Configuration Guide*.

In the following procedure, the migrated version 3.0 agent instance uses a new agent profile named `Migratedv3.0Agent` 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, refer to the respective policy agent 3.0 guide.

▼ 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 web container files and directories.

2 **If necessary, set your `JAVA_HOME` environment variable to point to an installed JDK version 1.5 or later.**

3 **Stop the web container instance for the version 2.2 agent.**

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

5 **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 Sun Downloads site under View by Category, Identity Management, and then Policy Agents: `http://www.sun.com/download/index.jsp`

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

For example, if you downloaded and unzipped the version 3.0 Apache HTTP Server 2.0.x agent in the `/opt/v30agent` directory:

```
cd /opt/v30agent/web_agents/apache_agent/bin
```

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:**

```
/opt/v22agent/web_agents/apache_agent
```

In this example, `/opt/v22agent` is the directory where you downloaded and unzipped the version 2.2 agent.

The `agentadmin` program migrates the version 2.2 agent.

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

   `Agent_nnn` is the policy agent instance. For example: `Agent_001` or `Agent_002`

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

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

11. **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: `/tmp/amadminpw`

12. **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 -b Migratedv3.0Agent -t WebAgent -u amadmin
    -f /tmp/amadminpw -D ./OpenSSOAgentConfiguration.properties
    ```

    In this example:

    - *tools-zip-root* is the directory where you unzipped the `openssoAdminTools.zip` file.
    - `Migratedv3.0Agent` is the version 3.0 agent profile name.
    - `WebAgent` is the agent type for web agents. For a Java EE agent, the agent type is `J2EEAgent`.
    - `/tmp/amadminpw` is the path to the `amadmin` password 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.

13. **Restart the web container 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.

Using a Version 2.2 Policy Agent After Upgrading to OpenSSO Enterprise

In this scenario, you have upgraded an Access Manager 7.x deployment to OpenSSO Enterprise 8.0 but you want to use an existing version 2.2 policy agent. Considerations are:

- **Coexistence**. Version 2.2 and version 3.0 policy agents can coexist in the same OpenSSO Enterprise deployment.
- **Location of Agent Configuration Repository**. A version 2.2 agent must continue to store its configuration data locally in its `AMAgent.properties` file on the agent's host server. Therefore, because the version 2.2 agent configuration data is local to the agent, the OpenSSO Enterprise centralized agent configuration feature is not supported for a version 2.2 agent. To configure a version 2.2 agent, you must continue to edit the agent's `AMAgent.properties` file.
- **Deployment URI**. If you are configuring a version 2.2 policy agent with OpenSSO Enterprise, the default Primary Server Deployment URI (and Failover Server Deployment URI, if required by the agent) is `/opensso` rather than `/amserver`.
- **Agent Profile**. You can create a version 2.2 Java EE (formerly J2EE) or web agent profile in the OpenSSO Enterprise Administration Console under Access Control, *realm-name*, Agents, and 2.2 Agents. However, you must configure the agent by editing its `AMAgent.properties` file.
- **Realm Mode**. If the Access Manager deployment was configured for Realm Mode, the realm name notation is changed in the URL after the upgrade to OpenSSO Enterprise. For example, if the Access Manager realm name was `users`, the name will be `/users` in OpenSSO Enterprise. You will need to reconfigure a version 2.2 policy agent to use the new realm name; otherwise. the "No such Organization found" message will be returned.

For more information about version 2.2 policy agents, see:

- Sun Downloads site under View by Category, Identity Management, and then Policy Agents: `http://www.sun.com/download/index.jsp`
- Documentation: `http://docs.sun.com/coll/1322.1`

Sun Microsystems Related Information

Additional Sun Resources

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

- Sun 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, visit `http://sun.com/access`.

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.

Accessing Sun Resources Online

The docs.sun.com web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. Books are available as online files in PDF and HTML formats. Both formats are readable by assistive technologies for users with disabilities.

To access the following Sun resources, go to `http://www.sun.com`:

- Downloads of Sun products
- Services and solutions
- Support (including patches and updates)
- Training

- Research
- Communities (for example, Sun Developer Network)

Third-Party Web Site References

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 or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

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 the Feedback link. 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 book's title page or in the document's URL. For example, the part number of this book is 820-5019.

Revision History

Date (Revision)	Description of Changes
March 9, 2010 (820-5019-14)	Revised to fix issues 5648 and 5649 and CR 6887525.
August 19, 2009 (820-5019-13)	■ Removed references to Sun Java System Access Manager 6 2005Q1 (6.3), because upgrade from this release it not currently supported. ■ Revised to fix issues 3581 and 1791.
March 17, 2009 (820-5019-12)	Rewrite to close issues.
December 18, 2008 (820-5019-11)	■ Added the "Upgrading Multiple Instances of Access Manager" on page 25 section. ■ Clarified the `ssopre80upgrade` parameter for the OpenSSO 8.0 Enterprise upgrade directory and configuration directory in "Running the Pre-Upgrade Script" on page 15.
November 11, 2008 (820-5019-10)	Initial release.