Sun Federated Access Manager 8.0 Early Access (EA) Release Notes

Beta

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

Part No: 820–3745–05
August 19, 2008

080819@20795

Contents

Sun Federated Access Manager 8.0 Early Access (EA) Release Notes

Early Access (EA) release. Last updated August 19, 2008

Sun™ Federated Access Manager 8.0 is being developed as part of the OpenSSO project (`http://opensso.org/`) and is the commercial version of OpenSSO server. These *Release Notes* provide information about the Federated Access Manager/OpenSSO 8.0 EA release, including:

Getting Started With Federated Access Manager/OpenSSO 8.0

If you have not previously installed Federated Access Manager/OpenSSO 8.0, here are the basic steps to follow:

1. If necessary, install, configure, and start one of the "Supported Web Containers" on page 11.
2. Download and unzip the `opensso.zip` file from the OpenSSO project site: `https://opensso.dev.java.net/public/use/index.html`.

   If you need `fam.zip`, contact your Sun representative.
3. Deploy the `opensso.war` or `fam.war` file to the web container, using the web container administration console or deployment command.

Or, if supported by the container, simply copy the WAR file to the container's autodeploy directory.

4. Launch the Federated Access Manager/OpenSSO 8.0 Configurator using the specific web container command or by specifying the following URL in your browser:
protocol://*host*.*domain*:*port*/*deploy_uri*

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

5. Configure Federated Access Manager/OpenSSO 8.0 server using the Configurator. For any additional configuration, use the either Administration Console or the new `famadm` command-line utility.
6. To download a version 3.0 policy agent, see `https://opensso.dev.java.net/public/use/index.html`.

Documentation. The Federated Access Manager 8.0 EA documentation is available on the OpenSSO project site:

`https://opensso.dev.java.net/public/use/docs/fampdf/index.html`

Note: Although this documentation specifically refers to Federated Access Manager 8.0, including terms such as `FAM.zip`, `fam.war`, and `/fam`, the documentation also applies to OpenSSO server.

Check this site periodically to view the most up-to-date documentation.

What's New in This Release

Federated Access Manager/OpenSSO 8.0 includes features such as access management, federation management, and web services security that are found in earlier releases of Sun Java System Access Manager and Sun Java System Federation Manager. Federated Access Manager/OpenSSO 8.0 also includes the following new features:

- "New Features in Federated Access Manager/OpenSSO 8.0" on page 6
- "New Features in Version 3.0 Policy Agents" on page 9

New Features in Federated Access Manager/OpenSSO 8.0

- Simplified installation and configuration:
  - To install Federated Access Manager/OpenSSO 8.0, you simply deploy the `fam.war` or `opensso.war` file using the respective web container administration console or command-line utility. When you first access the server using the deployment URI (`/fam`

or `/opensso`), you are directed to the Configurator, which allows you to perform initial configuration tasks such as specifying administrator passwords and the configuration and user data stores.

- You can also create and deploy specialized WAR files for a distributed authentication UI server, console only, server only, and Identity Provider (IDP) Discovery Service deployments using the `fam.war` or `opensso.war` file.

- Centralized server and agent configuration data:
  - Federated Access Manager/OpenSSO 8.0 and version 3.0 policy agent configuration data is stored in a centralized configuration data repository. You specify configuration values using either the Federated Access Manager/OpenSSO 8.0 Administration Console or the new `famadm` command-line utility. You no longer need to set properties in the `AMConfig.properties` or `AMAgent.properties` files.
  - Many of the configuration properties are "hot swappable," which means you do not have to restart the web container after you modify a property.
  - The Embedded data store option allows you to store Federated Access Manager/OpenSSO 8.0 and version 3.0 policy agent configuration data transparently without having to install Sun Java System Directory Server.
- Federated Access Manager/OpenSSO 8.0 Administration Console Common Tasks:
  - Create SAMLv2 Providers. You can easily create a SAMLv2 hosted or remote Identity Provider (IDP) or Service Provider (SP).
  - Create a Fedlet. A Fedlet is a lightweight Service Provider (SP) implementation of SAMLv2 SSO protocols. A Fedlet allows an Identity Provider (IP) to enable an SP that does not have federation implemented. The SP simply adds the Fedlet to a Java web application and then deploys the application.
  - Test Federation Connectivity. You can test or troubleshoot new or existing federated deployments to determine if connections are being made successfully and to identify the source of any problems.
- New web containers are added, as described in "Supported Web Containers" on page 11.
- Simplified Web Services Security agents can be deployed on Glassfish and Sun Java System Application Server 9.1 using providers based on the JSR 196 SPI.
- WS-Federation supports the Identity Federation specification. Federated Access Manager/OpenSSO 8.0 specifically supports the WS-Federation Passive Requestor Profile.
- Support for XACML version 2.0 support is added, specifically for `XACMLAuthzDecisionQuery` and `XACMLAuthzDecisionStatement`, as specified in the SAML 2.0 profile of XACML v2.0.
- Secure Authentication and Attribute Exchange allows an application to provide user authentication and attribute information with secure transfers between IDP and SP applications.

- Multiple federation protocol hub allows an Federated Access Manager/OpenSSO 8.0 IDP to act as federation hub to perform single logout among different federation protocols (such as SAMLv2, ID-FF, and WS-Federation).
- SAMLv2 profile support includes IDP proxying, Affiliation, NameID mapping, ECP, Authentication Query, and Attribute Query.
- Security Token Service (STS) is available on "Supported Web Containers" on page 11.
- SAMLv2 assertion failover is supported.
- New command-line utility (`famadm`) can configure both Federated Access Manager/OpenSSO 8.0 server and version 3.0 policy agents.
- Integration with Sun Java System Identity Manager, SiteMinder, and Oracle Access Manager is added.
- Service Tags are supported.

  Federated Access Manager 8.0 is Service Tag enabled. To use Service Tags, you must first register your product. On the Federated Access Manager Admin Console, under `Common Tasks`, click `Register This Product`. To register, you need a Sun Online Account (SOA) or Sun Developer Network (SDN) account. If you do not have one of these accounts, you can get an account during the product registration process.

  For more information about Service Tags and Sun Connection, see `http://lv.sun.com/practice/services/sun_connect/index.jsp`.

  To check your inventory, use the Sun Inventory site: `https://inventory.sun.com/inventory/`
- Internationalization and localization changes include:
  - In addition to English, Federated Access Manager/OpenSSO 8.0 includes support for French, Spanish, German, Japanese, Korean, Simplified Chinese, and Traditional Chinese.
  - Localized files are bundled in the `opensso.war` or `fam.war` files by default (unlike Access Manager 7 2005Q4 and Access Manager 7.1, where localized files reside in separate localized packages).
- Unix, SecurID, and SafeWord authentication modules

  Note: The Unix, SecurID, and SafeWord authentication modules are available only in Federated Access Manager and not in OpenSSO server. The SecurID is not available in the EA release. If you need the Unix or SafeWord module, contact your Sun representative.
- Upgrade support includes:
  - Upgrade to Federated Access Manager 8.0 from Access Manager 6.3, 7.0, or 7.1 and Federation Manager 7.0
  - Policy agent upgrade to version 3.0 from version 2.2 agents

New Features in Version 3.0 Policy Agents

Sun is developing version 3.0 policy agents in conjunction with Federated Access Manager 8.0. The version 3.0 agents have the following new features and improvements:

- Centralized agent configuration

  The centralized agent configuration feature moves most of the agent configuration properties from the `AMAgent.properties` file to the Federated Access Manager central data repository. An agent administrator can then manage the multiple agent configurations from a central server location, using either the Federated Access Manager Administration Console or the `famadm` command-line utility. The agent administrator no longer needs to edit an agent's `AMAgent.properties` file.

  The centralized agent configuration feature separates the version 3.0 agent configuration data into two sets:

  - The properties required for the agent to start up and initialize itself are stored in the `FAMAgentBootstrap.properties` file locally on the server where the agent is installed. For example, the agent profile name and password used to access the Federated Access Manager server are stored in the bootstrap file.
  - The rest of the agent properties are stored either centrally in the Federated Access Manager data repository (centralized configuration option) or locally in the `FAMAgentConfiguration.properties` file (local configuration option).

- Agent types

  Version 3.0 agents are classified according to type: `J2EEAgent` or `WebAgent`.

- Agent groups

  You can assign version 3.0 agents of the same type (`J2EEAgent` or `WebAgent`) to an agent group. All agents in a group then selectively share a common set of configuration properties. Thus, the agent configuration and management is simplified, because an administrator can manage all of the agents within a group as a single entity.

  Although all agents in the same group can share the same properties, you might need to define some individual properties for an agent (for example, the notification URL or agent URI properties).

- More hot-swappable agent configuration properties

  Version 3.0 agents have more hot-swappable configuration properties. An administrator can change a hot-swappable configuration property value for an agent without having to restart the agent's deployment container for the new value to take effect. Properties in `FAMAgentBootstrap.properties` are not hot-swappable.

- One-level wildcard support in URL policy

  While the regular wildcard support applies to multiple levels in a resource, the one-level wildcard applies to only the level where it appears in a resource.

- Default J2EE agent installation option with minimal questions asked during the installation

Default or custom installation:

- **Default** (`agentadmin --install`): The `agentadmin` program displays a minimal number of prompts and uses default values for the other options. Use the default install option when the default option meet your deployment requirements.
- **Custom** (`agentadmin --custom-install`): The `agentadmin` program displays a full set of prompts, similar to the version 2.2 program. Use the custom install option when you want to specify values other than the default options.

- Option to create the agent profile for J2EE agents in the server during installation

  The 3.0 agent installer supports an option to create the agent profile in the Federated Access Manager server during the agent installation so you don't have to create the profile manually using the Federated Access Manager Console or `famadm` utility.

- Automated migration support

  You can migrate a version 2.2 agent to a version 3.0 agent using the `agentadmin` program with the `--migrate` option.

  Note: Federated Access Manager does not support version 2.1 policy agents.

Federated Access Manager/OpenSSO 8.0 Hardware and Software Requirements

Operating System (OS) Support

TABLE 1 Operating System (OS) Support

Operating System	Supported Web Containers
Solaris 10 OS on SPARC, x86, and x64 based systems	All "Supported Web Containers" on page 11
Solaris 9 OS on SPARC and x86 based systems	

TABLE 1 Operating System (OS) Support *(Continued)*

Operating System	Supported Web Containers
Red Hat Enterprise Linux 4 server (Base and Advanced Platform) Red Hat Enterprise Linux 5 (Base and Advanced Platform)	All "Supported Web Containers" on page 11 except Geronimo
Ubuntu 8.0.4	"Glassfish Application Server V2 UR1 and UR2" on page 13 only
Windows Server 2003 Standard Edition Windows Server 2003 Enterprise Edition Windows 2003 Server Datacenter Edition Windows Vista	All "Supported Web Containers" on page 11 except Geronimo
IBM AIX 5.3	"IBM WebSphere Application Server 6.1" on page 18 only

Supported Web Containers

Federated Access Manager/OpenSSO 8.0 supports the following web containers:

- "Sun Java System Application Server 9.1 Update 1 and Update 2" on page 11
- "Glassfish Application Server V2 UR1 and UR2" on page 13
- "Sun Java System Web Server 7.0 Update 3" on page 14
- "Apache Tomcat 5.5.x and 6.x" on page 14
- "BEA WebLogic Server 9.2 MP2" on page 15
- "BEA WebLogic Server 10" on page 16
- "Oracle Application Server 10g" on page 18
- "IBM WebSphere Application Server 6.1" on page 18
- "Apache Geronimo Application Server 2.1.1" on page 20
- "JBoss Application Server 4.x" on page 22

Sun Java System Application Server 9.1 Update 1 and Update 2

Pre-Deployment Tasks

In the Application Server 9.1 domain where you plan to deploy Federated Access Manager/OpenSSO 8.0server, change the following JVM options either using the Application Server administration console or CLI utility:

- Change `-Xmx512m` to `-Xmx1024m`.
- If necessary, change `-client` to `-server`.

If the Java Security Manager is enabled, add the following permissions to the `server.policy` file. After you edit the file, restart the web container.

```
grant {
permission java.net.SocketPermission "*", "listen,connect,accept,resolve";
permission java.util.PropertyPermission "*", "read, write";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "setFactory";
permission java.lang.RuntimePermission "accessClassInPackage.*";
permission java.util.logging.LoggingPermission "control";
permission java.lang.RuntimePermission "shutdownHooks";
permission javax.security.auth.AuthPermission "getLoginConfiguration";
permission javax.security.auth.AuthPermission "setLoginConfiguration";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "createLoginContext.*";
permission java.io.FilePermission "<<ALL FILES>>", "read,write,execute,delete";
permission java.util.PropertyPermission "java.util.logging.config.class", "write";
permission java.security.SecurityPermission "removeProvider.SUN";
permission java.security.SecurityPermission "insertProvider.SUN";
permission javax.security.auth.AuthPermission "doAs";
permission java.util.PropertyPermission "java.security.krb5.realm", "write";
permission java.util.PropertyPermission "java.security.krb5.kdc", "write";
permission java.util.PropertyPermission "java.security.auth.login.config", "write";
permission java.util.PropertyPermission "user.language", "write";
permission javax.security.auth.kerberos.ServicePermission "*", "accept";
permission javax.net.ssl.SSLPermission "setHostnameVerifier";
permission java.security.SecurityPermission "putProviderProperty.IAIK";
permission java.security.SecurityPermission "removeProvider.IAIK";
permission java.security.SecurityPermission "insertProvider.IAIK";
permission java.lang.RuntimePermission "setDefaultUncaughtExceptionHandler";
permission javax.management.MBeanServerPermission "newMBeanServer";
permission javax.management.MBeanPermission "*", "registerMBean";
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.management.MBeanTrustPermission "register";
permission java.lang.management.ManagementPermission "monitor";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission java.util.PropertyPermission "javax.xml.soap.MetaFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.MessageFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPConnectionFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPFactory", "write";
permission java.net.NetPermission "getProxySelector";
permission java.security.SecurityPermission "getProperty.authconfigprovider.factory";
permission java.security.SecurityPermission "setProperty.authconfigprovider.factory";
permission javax.security.auth.AuthPermission "doAsPrivileged";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
```

```
permission java.security.SecurityPermission "insertProvider.XMLDSig";
permission java.security.SecurityPermission "putProviderProperty.WSS_TRANSFORM";
permission java.security.SecurityPermission "insertProvider.WSS_TRANSFORM";
};
```

Glassfish Application Server V2 UR1 and UR2

Glassfish download locations are:

- Glassfish V2 UR1: https://glassfish.dev.java.net/downloads/v2ur1-b09d.html
- Glassfish V2 UR2: https://glassfish.dev.java.net/downloads/v2ur2-b04.html

Pre-Deployment Tasks

In the Glassfish domain where you plan to deploy Federated Access Manager/OpenSSO 8.0 server, change the following JVM options either using the Glassfish administration console or by editing the `domain.xml` file:

- Change `-client` to `-server`.
- Change `-Xmx512m` to `-Xmx1024m`.

If the Java Security Manager is enabled, add the following permissions to the `server.policy` file. After you edit the file, restart the web container.

```
grant {
permission java.net.SocketPermission "*", "listen,connect,accept,resolve";
permission java.util.PropertyPermission "*", "read, write";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "setFactory";
permission java.lang.RuntimePermission "accessClassInPackage.*";
permission java.util.logging.LoggingPermission "control";
permission java.lang.RuntimePermission "shutdownHooks";
permission javax.security.auth.AuthPermission "getLoginConfiguration";
permission javax.security.auth.AuthPermission "setLoginConfiguration";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "createLoginContext.*";
permission java.io.FilePermission "<<ALL FILES>>", "read,write,execute,delete";
permission java.util.PropertyPermission "java.util.logging.config.class", "write";
permission java.security.SecurityPermission "removeProvider.SUN";
permission java.security.SecurityPermission "insertProvider.SUN";
permission javax.security.auth.AuthPermission "doAs";
permission java.util.PropertyPermission "java.security.krb5.realm", "write";
permission java.util.PropertyPermission "java.security.krb5.kdc", "write";
permission java.util.PropertyPermission "java.security.auth.login.config", "write";
permission java.util.PropertyPermission "user.language", "write";
permission javax.security.auth.kerberos.ServicePermission "*", "accept";
permission javax.net.ssl.SSLPermission "setHostnameVerifier";
permission java.security.SecurityPermission "putProviderProperty.IAIK";
```

```
permission java.security.SecurityPermission "removeProvider.IAIK";
permission java.security.SecurityPermission "insertProvider.IAIK";
permission java.lang.RuntimePermission "setDefaultUncaughtExceptionHandler";
permission javax.management.MBeanServerPermission "newMBeanServer";
permission javax.management.MBeanPermission "*", "registerMBean";
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.management.MBeanTrustPermission "register";
permission java.lang.management.ManagementPermission "monitor";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission java.util.PropertyPermission "javax.xml.soap.MetaFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.MessageFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPConnectionFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPFactory", "write";
permission java.net.NetPermission "getProxySelector";
permission java.security.SecurityPermission "getProperty.authconfigprovider.factory";
permission java.security.SecurityPermission "setProperty.authconfigprovider.factory";
permission javax.security.auth.AuthPermission "doAsPrivileged";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
permission java.security.SecurityPermission "insertProvider.XMLDSig";
permission java.security.SecurityPermission "putProviderProperty.WSS_TRANSFORM";
permission java.security.SecurityPermission "insertProvider.WSS_TRANSFORM";
};
```

Sun Java System Web Server 7.0 Update 3

Note – Federated Access Manager/OpenSSO 8.0server supports Web Server 7.0 Update 3 only. Web Server 7.0 Update 1 and Web Server 7.0 Update 2 are **not** supported.

Pre-Deployment Tasks

Using the Web Server 7.0 administration console or CLI, set the JVM heap size option from the default `-Xms128M -Xmx256M` to `-Xms256M -Xmx512M`.

Apache Tomcat 5.5.x and 6.x

Note – Do **not** use Tomcat 5.5.26 or Tomcat 6.0.16 for the Federated Access Manager/OpenSSO 8.0 EA release.

Pre-Deployment Tasks

For both Tomcat 5.5.x and Tomcat 6.x, set the `-Xmx` JVM option to `-Xmx1024m`.

BEA WebLogic Server 9.2 MP2

WebLogic Server 9.2 MP2 is supported on the operating systems shown on the following site:

http://e-docs.bea.com/
platform/suppconfigs/configs92/92_over/overview.html#1122259

Pre-Deployment Tasks

In the *bea_home*/user_projects/domains/*domain_name*/bin/setDomainEnv.sh script, add the click.mode=debug system property using JVM_OPTIONS.

On Windows systems set the JVM_OPTIONS in setDomainEnv.cmd. For example:

```
set JVM_OPTIONS=-Dclick.mode=debug
```

If you are using the Security Token Service (STS), set the MaxPermSize JVM option to a minimum value of 128 MB. For example:

```
-XX:MaxPermSize=128M
```

If the Java Security Manager is enabled, add the following permissions to the weblogic.policy file:

```
grant {
permission java.net.SocketPermission "*", "listen,connect,accept,resolve";
permission java.util.PropertyPermission "*", "read, write";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "setFactory";
permission java.lang.RuntimePermission "accessClassInPackage.*";
permission java.util.logging.LoggingPermission "control";
permission java.lang.RuntimePermission "shutdownHooks";
permission javax.security.auth.AuthPermission "getLoginConfiguration";
permission javax.security.auth.AuthPermission "setLoginConfiguration";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "createLoginContext.*";
permission java.io.FilePermission "<<ALL FILES>>", "read,write,execute,delete";
permission java.util.PropertyPermission "java.util.logging.config.class", "write";
permission java.security.SecurityPermission "removeProvider.SUN";
permission java.security.SecurityPermission "insertProvider.SUN";
permission javax.security.auth.AuthPermission "doAs";
permission java.util.PropertyPermission "java.security.krb5.realm", "write";
permission java.util.PropertyPermission "java.security.krb5.kdc", "write";
permission java.util.PropertyPermission "java.security.auth.login.config", "write";
permission java.util.PropertyPermission "user.language", "write";
permission javax.security.auth.kerberos.ServicePermission "*", "accept";
permission javax.net.ssl.SSLPermission "setHostnameVerifier";
permission java.security.SecurityPermission "putProviderProperty.IAIK";
```

```
permission java.security.SecurityPermission "removeProvider.IAIK";
permission java.security.SecurityPermission "insertProvider.IAIK";
permission java.lang.RuntimePermission "setDefaultUncaughtExceptionHandler";
permission javax.management.MBeanServerPermission "newMBeanServer";
permission javax.management.MBeanPermission "*", "registerMBean";
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.management.MBeanTrustPermission "register";
permission java.lang.management.ManagementPermission "monitor";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission java.util.PropertyPermission "javax.xml.soap.MetaFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.MessageFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPConnectionFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPFactory", "write";
permission java.net.NetPermission "getProxySelector";
permission java.security.SecurityPermission "getProperty.authconfigprovider.factory";
permission java.security.SecurityPermission "setProperty.authconfigprovider.factory";
permission javax.security.auth.AuthPermission "doAsPrivileged";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
permission java.security.SecurityPermission "insertProvider.XMLDSig";
permission java.security.SecurityPermission "putProviderProperty.WSS_TRANSFORM";
permission java.security.SecurityPermission "insertProvider.WSS_TRANSFORM";
};
```

BEA WebLogic Server 10

WebLogic Server 10 is supported on the operating systems shown on the following site:

`http://e-docs.bea.com/platform/suppconfigs/configs100/100_over/overview.html#1122259`

Pre-Deployment Tasks

In the *bea_home*`/user_projects/domains/`*domain_name*`/bin/setDomainEnv.sh` script, add the `click.mode=debug` system property using `JVM_OPTIONS`.

On Windows systems set the `JVM_OPTIONS` in `setDomainEnv.cmd`. For example:

```
set JVM_OPTIONS=-Dclick.mode=debug
```

If you are using the Security Token Service (STS), set the `MaxPermSize` JVM option to a minimum value of 128 MB. For example:

```
-XX:MaxPermSize=128M
```

If the Java Security Manager is enabled, add the following permissions to the `weblogic.policy` file:

```
grant {
permission java.net.SocketPermission "*", "listen,connect,accept,resolve";
permission java.util.PropertyPermission "*", "read, write";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "setFactory";
permission java.lang.RuntimePermission "accessClassInPackage.*";
permission java.util.logging.LoggingPermission "control";
permission java.lang.RuntimePermission "shutdownHooks";
permission javax.security.auth.AuthPermission "getLoginConfiguration";
permission javax.security.auth.AuthPermission "setLoginConfiguration";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "createLoginContext.*";
permission java.io.FilePermission "<<ALL FILES>>", "read,write,execute,delete";
permission java.util.PropertyPermission "java.util.logging.config.class", "write";
permission java.security.SecurityPermission "removeProvider.SUN";
permission java.security.SecurityPermission "insertProvider.SUN";
permission javax.security.auth.AuthPermission "doAs";
permission java.util.PropertyPermission "java.security.krb5.realm", "write";
permission java.util.PropertyPermission "java.security.krb5.kdc", "write";
permission java.util.PropertyPermission "java.security.auth.login.config", "write";
permission java.util.PropertyPermission "user.language", "write";
permission javax.security.auth.kerberos.ServicePermission "*", "accept";
permission javax.net.ssl.SSLPermission "setHostnameVerifier";
permission java.security.SecurityPermission "putProviderProperty.IAIK";
permission java.security.SecurityPermission "removeProvider.IAIK";
permission java.security.SecurityPermission "insertProvider.IAIK";
permission java.lang.RuntimePermission "setDefaultUncaughtExceptionHandler";
permission javax.management.MBeanServerPermission "newMBeanServer";
permission javax.management.MBeanPermission "*", "registerMBean";
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.management.MBeanTrustPermission "register";
permission java.lang.management.ManagementPermission "monitor";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission java.util.PropertyPermission "javax.xml.soap.MetaFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.MessageFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPConnectionFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPFactory", "write";
permission java.net.NetPermission "getProxySelector";
permission java.security.SecurityPermission "getProperty.authconfigprovider.factory";
permission java.security.SecurityPermission "setProperty.authconfigprovider.factory";
permission javax.security.auth.AuthPermission "doAsPrivileged";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
permission java.security.SecurityPermission "insertProvider.XMLDSig";
permission java.security.SecurityPermission "putProviderProperty.WSS_TRANSFORM";
permission java.security.SecurityPermission "insertProvider.WSS_TRANSFORM";
};
```

Oracle Application Server 10g

No pre-installation tasks are required.

IBM WebSphere Application Server 6.1

Pre-Deployment Tasks

Adding genericJvmArguments

Add the `genericJvmArguments` using the WebSphere Admin Console or by editing the `server.xml` file:

1. Open the following file:

   install_root`/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/`*cell*`/nodes/`*node*`/servers/`*server*`/server.xml`

2. Find the `jvmEntries` element.
3. Add the following `genericJvmArguments` and save the file:

   ```
   genericJvmArguments="-DamCryptoDescriptor.provider=IBMJCE
   -DamKeyGenDescriptor.provider=IBMJCE"
   ```

4. Restart WebSphere 6.1 Application Server.

Adding Permissions to the server.policy File

In the *install_root*`/IBM/WebSphere/AppServer/profiles/AppSrv01/properties/server.policy` file, add the following permissions. After you edit the file, restart the web container.

```
grant {
permission java.net.SocketPermission "*", "listen,connect,accept,resolve";
permission java.util.PropertyPermission "*", "read, write";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "setFactory";
permission java.lang.RuntimePermission "accessClassInPackage.*";
permission java.util.logging.LoggingPermission "control";
permission java.lang.RuntimePermission "shutdownHooks";
permission javax.security.auth.AuthPermission "getLoginConfiguration";
permission javax.security.auth.AuthPermission "setLoginConfiguration";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "createLoginContext.*";
permission java.io.FilePermission "<<ALL FILES>>", "read,write,execute,delete";
permission java.util.PropertyPermission "java.util.logging.config.class", "write";
permission java.security.SecurityPermission "removeProvider.SUN";
permission java.security.SecurityPermission "insertProvider.SUN";
permission javax.security.auth.AuthPermission "doAs";
```

```
permission java.util.PropertyPermission "java.security.krb5.realm", "write";
permission java.util.PropertyPermission "java.security.krb5.kdc", "write";
permission java.util.PropertyPermission "java.security.auth.login.config", "write";
permission java.util.PropertyPermission "user.language", "write";
permission javax.security.auth.kerberos.ServicePermission "*", "accept";
permission javax.net.ssl.SSLPermission "setHostnameVerifier";
permission java.security.SecurityPermission "putProviderProperty.IAIK";
permission java.security.SecurityPermission "removeProvider.IAIK";
permission java.security.SecurityPermission "insertProvider.IAIK";
permission java.lang.RuntimePermission "setDefaultUncaughtExceptionHandler";
permission javax.management.MBeanServerPermission "newMBeanServer";
permission javax.management.MBeanPermission "*", "registerMBean";
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.management.MBeanTrustPermission "register";
permission java.lang.management.ManagementPermission "monitor";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission java.util.PropertyPermission "javax.xml.soap.MetaFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.MessageFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPConnectionFactory", "write";
permission java.util.PropertyPermission "javax.xml.soap.SOAPFactory", "write";
permission java.net.NetPermission "getProxySelector";
permission java.security.SecurityPermission "getProperty.authconfigprovider.factory";
permission java.security.SecurityPermission "setProperty.authconfigprovider.factory";
permission javax.security.auth.AuthPermission "doAsPrivileged";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
permission java.security.SecurityPermission "insertProvider.XMLDSig";
permission java.security.SecurityPermission "putProviderProperty.WSS_TRANSFORM";
permission java.security.SecurityPermission "insertProvider.WSS_TRANSFORM";
};
```

Running the JSP Compiler With WebSphere Application Server 6.1

WebSphere Application Server 6.1 has an Eclipse-based JSP compiler that uses JDT (Java Development Tooling) and the AST (Abstract Syntax Tree) parser. For information about parsing and generating the Java code, see:

`http://www-128.ibm.com/developerworks/opensource/library/os-ast/`

If you plan to deploy Federated Access Manager/OpenSSO 8.0 in the WebSphere Application Server 6.1 Admin Console, the source level for the JSPs must be set to 15 (JVM 1.5). The default is 13 (JVM 1.3), which causes compilation problems. See issue "1977: SAMLv2 sample `configure.jsp` files fail on WebSphere Application Server 6.1" on page 26.

This compiler depends on some of the user `env` settings, and if they are not propagated during the compiler initialization, the compiler can fail to initialize properly. There are two workarounds to this issue:

- Install WebSphere Application Server 6.1 as a non-`root` user, and the installation process should work as expected.

  or

- Modify your web archive descriptor for the JDK compiler. Edit your `ibm-web-ext.xmi` file under the web module deployment directory and add a line similar to:

  ```
  <jspAttributes xmi:id="JSPAttribute_nnnnn" name="useJDKCompiler"
  value="true"/>
  ```

  where *nnnnn* is any unique number.

Using the `famadm` and `ampassword` Utilities

Before you run the `setup` script to install the tools and scripts, modify the `setup` script. Before `-cp ...` in the last line, insert:

```
-D"amCryptoDescriptor.provider=IBMJCE" -D"amKeyGenDescriptor.provider=IBMJCE"
```

After you run the `setup` script to install the tools and before you run `famadm`, add the following items to the `famadm` script:

- Add `xalan.jar` to the `classpath` after `openfedlib.jar`. For example:

  ```
  ${TOOLS_HOME}/lib/xalan.jar
  ```

- Add the following items before `com.sun.identity.cli.CommandManager` and `com.sun.identity.tools.bundles.Main`:

  ```
  -D"amKeyGenDescriptor.provider=IBMJCE"
  -D"amCryptoDescriptor.provider=IBMJCE"
  ```

After you run the `setup` script and before you run `ampassword`, add the following items to the `ampassword` script before `com.iplanet.services.ldap.ServerConfigMgr` and `com.sun.identity.tools.bundles.Main`:

```
-D"amCryptoDescriptor.provider=IBMJCE" -D"amKeyGenDescriptor.provider=IBMJCE"
```

Apache Geronimo Application Server 2.1.1

Note – Federated Access Manager/OpenSSO 8.0 server supports Geronimo Application Server 2.1.1 with Tomcat on Solaris systems only.

Pre-Deployment Task

Modify the `/geronimo-tomcat6-jee5-2.0.2/bin/geronimo.sh` file by adding `-X:MaxPermSize=512M`, as shown in the following start block:

```
elif [ "$1" = "start" ] ; then
shift
touch "$GERONIMO_OUT"
$START_OS_CMD "$_RUNJAVA" $JAVA_OPTS $GERONIMO_OPTS \
$JAVA_AGENT_OPTS \
-Dorg.apache.geronimo.base.dir="$GERONIMO_BASE" \
-Djava.endorsed.dirs="$ENDORSED_DIRS" \
-Djava.io.tmpdir="$GERONIMO_TMPDIR" \
-XX:MaxPermSize=512M \
-jar "$GERONIMO_HOME"/bin/server.jar $LONG_OPT "$@" \
>> $GERONIMO_OUT 2>&1 &
echo "" echo "Geronimo started in background. PID: $!"
if [ ! -z "$GERONIMO_PID" ]; then echo $! > $GERONIMO_PID
fi
```

To deploy the Federated Access Manager/OpenSSO 8.0 WAR file on Geronimo, you must provide a deployment plan file either inside or outside of the WAR file. If placed inside the WAR file, name the plan `geronimo-web.xml` and place the file in `WEB-INF` directory. If placed outside of the WAR file, the plan file can be named otherwise. Here is a sample plan file:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://geronimo.apache.org/xml/ns/j2ee/web-1.2">
<environment>
<moduleId>
<groupId>sun</groupId>
<artifactId>FAM</artifactId>
<version>8.0</version>
<type>war</type>
</moduleId>
</environment>
<context-root>/fam1</context-root>
</web-app>
```

In the above example, the WAR file is deployed at:

```
geronimo-tomcat6-jee5-2.0.2/repository/sun/FAM/8.0/FAM-8.0.war
```

The web application is deployed at *protocol*`://`*server*`:`*port*`/fam1`. You can change the deployment plan depending on your deployment scenario.

Notes:

- Geronimo console URL: *protocol*`://`*server*`:8080/console/portal/welcome`
- Default user name and password: `system/manager`
- To start the Geronimo server:`/geronimo-tomcat6-jee5-2.0.2/bin/geronimo.sh start`
- To stop the Geronimo server: `/geronimo-tomcat6-jee5-2.0.2/bin/geronimo.sh stop`

JBoss Application Server 4.x

Federated Access Manager/OpenSSO 8.0 server supports only the Exploded Deployment on JBoss Application Server 4.x. For more information see:

`http://wiki.jboss.org/wiki/Wiki.jsp?page=ExplodedDeployment`

If you are using the Security Token Service (STS), set the `MaxPermSize` JVM option to a minimum value of 128 MB. For example:

```
-XX:MaxPermSize=128M
```

To deploy Federated Access Manager/OpenSSO 8.0 server on JBoss Application Server 4.x:

1. Create a subdirectory under *JBOSS_HOME*`/server/`*instance*`/deploy/`*name_of_war_file*. For example:

   ```
   # mkdir /opt/jboss-4.2.2.GA/server/fam/deploy/fam.war
   ```

2. Explode the `opensso.war` or `fam.war` file in this new directory. For example: Your don't need to restart the container, because JBoss will automatically hot-deploy it.

   ```
   # cd /opt/jboss-4.2.2.GA/server/fam/deploy/opensso.war
   # jar xvf /tmp/opensso.war
   ```

3. Point your browser to `http://`*host*`.`*domain*`:`*port*`/opensso` or `http://`*host*`:`*port*`/fam`and start configuring Federated Access Manager/OpenSSO 8.0server.
4. The Federated Access Manager/OpenSSO 8.0 Configurator will write a bootstrap file in your home directory. For example:

   ```
   /AccessManager/AMConfig_opt_jboss-4.2.2.GA_server_fam_._deploy_opensso.war_
   ```

JDK Requirements

TABLE 2 JDK Requirements

Federated Access Manager/OpenSSO 8.0	Supported JDK Version
Server	JDK 1.5.x or 1.6.x 64-bit JVM on supported web containers
Client (FAMSDK)	JDK 1.4.x, 1.5.x. or JDK 1.6.x

Data Store Requirements

TABLE 3 Data Store Requirements

Data Store Type	Supported Data Stores
Configuration data store (also referred to as the Service Management data store)	■ Sun Java System Directory Server 5.2, 6.0, 6.2, and 6.3 ■ Federated Access Manager (embedded store)
User data store	■ Sun Java System Directory Server 5.2, 6.0, 6.2, and 6.3 ■ Federated Access Manager (embedded store) **Note**: The Federated Access Manager user data store is supported only for prototype, proof of concept (POC), or developer deployments that have a small set of users. It is **not** supported for production deployments. ■ Microsoft Active Directory 2003 on Windows Server 2003 R2 ■ IBM Tivoli Directory Server 6.1

Hardware Requirements

TABLE 4 Federated Access Manager Hardware Requirements

Component	Requirement
RAM	Prototype or developer deployment: 1 GB Production deployment: 4 GB recommended
Disk space	For server with console, server only, or console only deployment: ■ 512 MB for Federated Access Manager binary files and configuration data ■ 2 GB for log files, including container log files For client SDK deployment: ■ 100 MB minimum ■ 1 GB recommended for debug logs, if debug level (`com.iplanet.services.debug.level`) is set to `Message`

Federated Access Manager Supported Browsers

TABLE 5 Federated Access Manager Supported Browsers

Browser	Platform
Firefox 1.0.7 and 1.5	Windows XP
	Windows 2000
	Solaris OS, versions 9 and 10
	Red Hat Linux 4 and 5
Microsoft Internet Explorer 7	Windows XP and Windows 2003
Microsoft Internet Explorer 6.0 SP2	Windows XP
Microsoft Internet Explorer 6.0 SP1	Windows 2000
Mozilla 1.7.12	Solaris OS, versions 9 and 10
	Windows XP
	Windows 2000
	Red Hat Linux 4 and 5
Netscape Communicator 8.0.4	Windows XP
	Windows 2000
Netscape Communicator 7.1	Solaris OS, versions 9 and 10

Federated Access Manager/OpenSSO 8.0 Issues

For more information about Federated Access Manager/OpenSSO 8.0 issues, see:

`https://opensso.dev.java.net/servlets/ProjectIssues`

830: ID-FF schema metadata is not backward compatible

If you are upgrading from Access Manager 6.3, 7.0, or 7.1 to Federated Access Manager/OpenSSO 8.0, ID-FF profiles do not work unless you also migrate the Federated Access Manager/OpenSSO 8.0 schema.

Workaround. Migrate the Federated Access Manager/OpenSSO 8.0 schema before you try the ID-FF profiles. For more information, see "Upgrading to Federated Access Manager/OpenSSO 8.0" on page 32.

1781: `amadmin` login fails for non data store authentication

If you change the authentication module for the root realm to anything besides `DataStore`, `amadmin` will not be able to log into the Console.

Workaround. Log in using `http://host.domain/deployurl/UI/Login?module=DataStore`.

1890: Displaying all users from a filtered role with default filter is blank

In the Federated Access Manager Console, displaying all users from a filtered role with the default filter where number of users is greater the size limit is blank.

1977: SAMLv2 sample `configure.jsp` files fail on WebSphere Application Server 6.1

On a WebSphere Application Server 6.1 instance, the `/sample/saml2/sp/configure.jsp` and `/sample/saml2/idp/configure.jsp` files fail to compile. The `configure.jsp` files require JDK 1.5, but the JDK source level for JSP files is set to JDK 1.3 on WebSphere Application Server 6.1.

Workaround: Edit the JSP engine configuration parameters to set the JDK source level to 1.5:

1. Open the `WEB-INF/ibm-web-ext.xmi` file.

   JSP engine configuration parameters are stored either in a web module's configuration directory or in a web module's binaries directory in the `WEB-INF/ibm-web-ext.xmi` file:

   Configuration directory. For example:

   `{WAS_ROOT}/profiles/`*profilename*`/config/cells/`*cellname*`/applications/`*enterpriseappname*`/deployments/`*deployedname*`/webmodulename/`

   Binaries directory, if an application was deployed into WebSphere Application Server with the flag "Use Binary Configuration" flag set to `true`. For example:

   `{WAS_ROOT}/profiles/`*profilename*`/installedApps/`*nodename*`/`*enterpriseappname*`/`*webmodulename*`/`
2. Delete the `compileWithAssert` parameter by either deleting the statement from the file or enclosing the statement with comment tags (<!– and –>).
3. Add the `jdkSourceLevel` parameter with the value of 15. For example:

   ```
   <jspAttributes xmi:id="JSPAttribute_1" name="jdkSourceLevel" value="15"/>
   ```

   Note: The integer (`_1`) in `JSPAttribute_1` must be unique within the file.
4. Save the `ibm-web-ext.xmi` file.
5. Restart the application.

For more information about the `jdkSourceLevel` parameter as well as other JSP engine configuration parameters, see:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/topic/com.ibm.websphere.nd.doc/info/ae/ae/rweb_jspengine.html

2182: Multiple server deployment with embedded configuration data store fails on Application Server 9.1

If you deploy two instances of Federated Access Manager on different host servers using the embedded configuration data store, the configuration of the second Federated Access Manager instance fails for the configuration data store.

Workaround. Add the `-Dcom.sun.enterprise.server.ss.ASQuickStartup=false` JVM option using either the Application Server Admin Console or by editing the `domain.xml` file. For example, in the `domain.xml` file, add the following entry:

```
<jvm-options>-Dcom.sun.enterprise.server.ss.ASQuickStartup=false</jvm-options>
```

This problem can also occur on Glassfish. For more information, see http://glassfish.dev.java.net/issues/show_bug.cgi?id=5321.

After you add the JVM option, restart the Application Server or Glassfish domain.

2222: Password reset and account lockout services report notification errors

Federated Access Manager/OpenSSO 8.0 submits email notifications using the unqualified sender name, `Identity-Server`, which returns error entries in the logs.

Workaround. Change the sender name from `Identity-Server` to `Identity-Server@hostname.domainname` in the following files:

- In `amPasswordResetModuleMsgs.properties`, change `fromAddress.label`.

- In `amAuth.properties`, change `lockOutEmailFrom`.

2348: Document Distributed Authentication UI server support

The Federated Access Manager/OpenSSO 8.0 Distributed Authentication UI server component works only with Federated Access Manager/OpenSSO 8.0. The following scenarios are not supported:

- Distributed Authentication UI server 7.0 or 7.1 with a Federated Access Manager/OpenSSO 8.0 server
- Federated Access Manager/OpenSSO 8.0 Distributed Authentication UI server with an Access Manager 7.0 or 7.1 server

2381: Access Manager Roles policy subject is supported only with AMSDK data store

The Access Manager Roles policy subject is supported only with the AMSDK data store. By default, this subject is disabled in the policy configuration. Therefore, enable the Access Manager Roles policy subject only if the data store type is configured to use the AMSDK plug-in.

2661: `logout.jsp` did not compile on WebSphere Application Server 6.1

The `logout.jsp` file requires JDK 1.5, but the JDK source level for JSP files is set to JDK 1.3 on IBM WebSphere Application Server 6.1.

Workaround. See the workaround for "1977: SAMLv2 sample `configure.jsp` files fail on WebSphere Application Server 6.1" on page 26.

2690: STS samples do not work with JDK 1.6

If you are using JDK 1.6, accessing the Security Token Service (STS) samples throws an exception.

2809: Windows `setup.bat` script failed to configure the `famadm` utility when using IBM JDK

When you run the `setup.bat` script to install the `famadm` utility and other Federated Access Manager/OpenSSO 8.0 tools on Windows systems, the following message is returned:

```
Unable to locate JRE meeting specification 1.4+
```

This problem occurs then the `JAVA_HOME` environment variable is set to the IBM JDK.

Workaround: Download Sun JDK 1.5 or later and set `JAVA_HOME` to that download directory.

2827: Configuring a site does not add the second server to the site

Session failover configuration does not add the second Federated Access Manager instance to the assigned servers list.

Workaround. Use the Federated Access Manager Console or `famadm` utility to manually add the second server instance to the servers list.

2833: Standalone stock quote sample does not work

The standalone stock quote sample does not work if you choose the Security Token Service (STS) as the security mechanism for the web service consumer (WSC).

2869: Remote Federated Access Manager/OpenSSO 8.0 configuration data store must be running

If you are using a remote Sun Java System Directory Server to store the Federated Access Manager/OpenSSO 8.0 configuration data, Directory Server must be running before you start the Federated Access Manager/OpenSSO 8.0 instance.

2883: Web Server 7.0 policy agent causes `OutOfMemoryError` exception in SSL mode

If the version 3.0 policy agent for Web Server 7.0 is configured with Federated Access Manager 8.0 in SSL mode, the Web Server container that is hosting the policy agent throws an `OutOfMemoryError` exception. This problem occurs because the minimum JVM heap size (`ms`) is set to a large value in the `server.xml` file, which is causing the C-Heap space to run out of memory.

Workaround: Set the minimum heap size to a lower value such as 512 MB, so that the rest of the heap space can be used by the C-Heap. For example, in the Web Server 7.0 `server.xml` file, set the minimum heap size as follows:

```
<jvm-options>-Xms512M -Xmx1024M </jvm-options>
```

Or, if you prefer set the minimum JVM heap size in the Web Server 7.0 Administration Console.

After you set the minimum JVM heap size, restart the Web Server 7.0 instance.

2905: `jss4.jar` entry is missing in the `famadm` classpath

After running the `setup` script for the `famadm` utility, trying to run `famadm` returns a `NoClassDefFoundError` error. This problem occurs for an upgraded Federated Access Manager/OpenSSO 8.0 instance.

Workaround. To use JSS, add `jss4.jar` to the `classpath` and set the `LD_LIBRARY_PATH` environment variable. (If you are using the default JCE, `jss4.jar` is not required to be in the `classpath`.)

2967: Adding new WebSphere Application Server instance to an existing site fails

When configuring a site with multiple Federated Access Manager instances behind a load balancer, adding a WebSphere Application Server 6.1 instance to the site fails during configuration.

2973: SafeWord authentication returns an error for WebSphere Application Server

With Federated Access Manager deployed on WebSphere Application Server 6.1 on Windows Server 2003 R2 x64 Edition, an attempt to perform authentication with a SafeWord authentication module instance returns a 500 error in the client browser.

2994: Linux C SDK has SPARC libraries

The C SDK samples cannot be built on Linux systems following the instructions in the README file because the Linux C SDK archive `agent-csdk-linux.tar.gz` includes a SPARC-based library (`libamsdk.so.3`).

3054: Session created by `famadm` CLI is not destroyed in a site deployment

In a multiple server Federated Access Manager deployment configured for session failover, a remote session created by `famadm` is not destroyed after a subcommand such as `create-realm` or `create-identity` executes.

Workaround. Make sure that sticky sessions are configured for the Federated Access Manager site.

3065: Same context ID is used for all users in ID-FF log records

All ID-FF log records have same the context (or login) ID, even if they are for different users.

3071: Top-level administrator login denied if session database is down

If two Federated Access Manager instances are configured for session failover, `amadmin` is denied access when the session database is down. For example, if you issue the `amsfo stop` command for each Federated Access Manager server in a session failover deployment, `amadmin` cannot log in to the console and a message is returned stating the maximum sessions limit was reached or the session quota has been exhausted.

This problem occurs because the Federated Access Manager `Deny user login when session repository is down` option is `YES` and the `Enable Quota Constraints` option is `ON`.

Workaround. Use the directory server console or a utility such as `ldapmodify` to change the value of the `iplanet-am-session-deny-login-if-db-is-down` attribute to `NO`.

3077, 3079, 2968: Certificate authentication fails if OCSP or LDAP checking is enabled

Certificate authentication fails as follows:

- LDAP checking is enabled with an `LDAPS` directory server.

  or

- OCSP checking is enabled.

3145: Federation fails in common domain deployment

Consider this scenario:

1. Set up a deployment with two IDPs, one SP, and one IDP proxy using `https` and `ldaps`.
2. Create users on each site.
3. Change cookie names on all three machines.
4. Run the ID-FF sample from the SP
5. Log in locally on the SP.
6. Click Federate, select one IDP, and submit.

An IDP login page is not displayed, and a `NullPointerException` (NPE) is thrown on the IDP side.

3165: Cannot import affiliation metadata using the `famadm` utility

After creating the affiliation metadata using the `create-metadata-template` option, trying to import the data using the `famadm import-entity` command does not work, and no errors are returned.

3203: ID-FF IDP proxy doesn't work as expected

Consider the scenario where one SP is deployed with a proxy and two IDPs. The SP user is federated to the proxy user, and then the proxy user is federated to the IDP users on the respective IDP. During an attempt to do single sign-on, the proxy doesn't redirect to the IDP. The proxy simply behaves as an IDP and stops at that point.

Upgrading to Federated Access Manager/OpenSSO 8.0

Upgrading to Federated Access Manager 8.0 is supported from Access Manager 6.3, 7.0, and 7.1. This section describes the basic steps to upgrade an Access Manager instance.

To upgrade to Federated Access Manager 8.0:

1. Login to the existing Access Manager 6.3, 7.0, or 7.1 instance to make sure that a valid instance exists.
2. Perform these pre-upgrade steps as needed:
   - Update Directory Server to a supported version.

- Update the Web container to a supported version.
- If you are on a Solaris system, make sure the version is Solaris 10 or higher.
- Backup the Access Manager or Federation Manager DIT schema.
- Collect the configuration data required for Step 10 (such as the encryption key and `amAdmin` user password).

3. Get the Federated Access Manager 8.0 ZIP file (`FAM.zip`).
4. Extract `FAM.zip` in a directory. For example: *zip_root*.
5. Create a staging directory and copy the `fam.war` file from the *zip_root*`/fam/deployable-war` to the new directory.
6. Run the pre-upgrade script (`fampre80upgrade`) to setup up the system for the upgrade.
7. Undeploy any Access Manager Web applications.
8. Create a new Federated Access Manager 8.0 WAR file in the staging directory:
   - Apply any customizations (such as auth, console UI, or samples).
   - Add any plug-in customizations.

   Note: The name of the new WAR file should match the URI of the Access Manager instance you are upgrading. For example, if the URI was `amserver`, the WAR should be named `amserver.war`.
9. Deploy the new WAR file. Use the same host and port number as the Access Manager instance you are upgrading.
10. Configure the WAR file using the Configurator. Select Custom Configuration and provide the same values as those used for Access Manager instance you are upgrading:
    - `amAdmin` password
    - `amldapuser` password (Agent Password in the Configurator)
    - Directory Server information (such as host and port)
    - Encryption key (from the `AMConfig.properties` file)
11. Update/migrate the Access Manager schema to Federated Access Manager 8.0 by running the `famupgrade` script in the *zip_root*`/fam/upgrade/scripts` directory.
12. Restart the Federated Access Manager web container.

Next Steps

Access Federated Access Manager 8.0 using *protocol*`://`*host*`.`*domain*`:`*port*`/`*deployuri*. For example:

```
http://famhost.example.com:8080/fam
```

Deprecation Notifications and Announcements

- The Service Management Service (SMS) APIs (`com.sun.identity.sm` package) and SMS model will not be included in a future Federated Access Manager release.
- The Unix authentication module and the Unix authentication helper (`amunixd`) will not be included in a future Federated Access Manager release.
- The *Sun Java System Access Manager 7.1 Release Notes* stated that the Access Manager `com.iplanet.am.sdk` package, commonly known as the Access Manager SDK (AMSDK), and all related APIs and XML templates will not be included in a future Federated Access Manager/OpenSSO 8.0 release. Migration options are not available now and are not expected to be available in the future. Sun Java System Identity Manager provides user provisioning solutions that you can use instead of the AMSDK. For more information about Identity Manager, see `http://www.sun.com/software/products/identity_mgr/index.jsp`.

How to Report Problems and Provide Feedback

If you have questions or issues with Federated Access Manager/OpenSSO 8.0, send an e-mail to Sun: `opensso.eafeedback@sun.com`

If you are requesting help for a problem, please include the following information:

- 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 Federated Access Manager/OpenSSO 8.0 version, including any patches or other software that might be affecting the problem
- Steps to reproduce the problem
- Any error logs or core dumps

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.

Revision History

TABLE 6 Revision History

Date	Description of Changes
August 4, 2008	Initial Early Access (EA) release
August 13, 2008	Added new issues
August 19, 2008	Added new issues