BEA AquaLogic Enterprise Security™®

How Tos

Version: 3.0
Document Revised: December 2007

Silent Mode Installations

SSL for Production Environments

Failover and System Reliability

Performance Statistics

Host Name or IP Address Change

Configuring SSL in the Web Services SSM

Database Password Changes

Resetting the ALES Administrator Password

Resource Discovery

Running the Java SSM in Java Development Environments

Debugging Policies

CHAPTER 1

Silent Mode Installations

When the Administration Server is installed, its configuration data is stored in an XML file. This file may be used to perform ‘silent installs’ of the server on other machines.

Entries in the configuration file correspond to the responses entered during a normal install. They can be modified as needed on the new machine.

Modify the Configuration File

To modify the configuration file for a silent install of the Administration Server:

1. Make a copy of the configuration file and open it in an editor. The file path is *BEA_HOME*`/ales30-admin/config/silent_install_admin.xml`.
2. Use Table 1-1 to modify the installation parameters. These are specified in XML syntax as name/value pairs. The values that can be modified are in the **value=** field. For example, in the entry below, you could change the directory name:

   ```
   <data-value name="USER_INSTALL_DIR" value="C:\bea\ales30-admin" />
   ```

Table 1-1 Silent Installation Configuration File

Data-Value Name	Description	Examples
BEAHOME	BEA_HOME directory	C:\bea
USER_INSTALL_DIR	Administration Server install directory	`C:\bea\ales30-admin`
SCM_INSTALL_DIR	SCM install directory.	`C:\bea\ales30-scm`

Table 1-1 Silent Installation Configuration File (Continued)

Data-Value Name	Description	Examples
WEB_SERVER_TYPE	Servlet container type being used.	weblogic81 weblogic92 weblogic10 tomcat
WEB_SERVER_DIR	Servlet container directory. **Note:** When using Tomcat, the directory name cannot contain spaces.	C:\bea\weblogic81
ADMIN_APP_PORT	Port for the servlet container's administration console.	7000
ADMIN_APP_SSL_PORT	SSL port for the Administration Server	7010
ADMIN_JAVA_HOME	Administration Server JDK	C:\bea\jrockit90_150_04
SCM_JAVA_HOME	SCM JDK	C:\bea\jrockit90_150_04
ENTERPRISE_DOMAIN_NAME	Must be **asi**.	
CERTIFICATE_DURATION	Years the security certificate remains in effect.	10
DATABASE_CLIENT	Database type	ORACLE92 ORACLE10 SYBASE125 SYBASE15 PointBase 5.1 MS SQL Server 2000 MS SQL Server 2005 DB2
DB_DRIVER_LOC	(MS SQL, Pointbase, DB2 only) Directory containing the database driver. **Note:** DB2 Driver license jar is shipped in separate jar. Append both jars and separate using OS-specific classpath separator.	
JDBC_URL	URL on which to reach the database	jdbc:sybase:Tds:ALESDB:5000

Table 1-1 Silent Installation Configuration File (Continued)

Data-Value Name	Description	Examples
JDBC_DRIVER	Java classname of the database driver.	com.sybase.jdbc3.jdbc.SybDri
DATABASE_LOGIN_ID	Username to access the database.	
DATABASE_LOGIN_PASS	Password for the above account. You must replace "*@db.password@*" with the actual value.	
KEY STORES: CA_KEY_PASS PEER_KEY_PASS TRUSTED_CA_KEY_PASS SCM_KEY_PASS SSM_KEY_PASS ADMIN_KEY_PASS	Key store passwords used for internal ALES component communications. If left blank, randomly generated passwords are used. Otherwise, provide a password for each entry.	
INSTALL_DB_SCHEMA	Specify whether or not to install the policy database schema.	no
SCM_INTERFACE_LIST	A comma-separated list of IP addresses of the network interfaces to which to bind the Service Control Manager.	169.254.25.129

Perform a Silent Installation

To run the Administration Server installation in silent mode:

1. Copy the modified configuration XML to a location on the machine.
2. Launch the install using the following command:

   Windows: ales300admin_win32.exe -mode=silent -silent_xml=<*path_file*>

   UNIX/Linux: ales300admin_solaris32.bin -mode=silent -silent_xml=<*path_file*>

   where

   <*path_file*>—path and file name of the configuration file

CHAPTER 2

SSL for Production Environments

ALES uses SSL for communications between the Administration Server, remote ALES components, and external clients. Installation of ALES includes demonstration certificates that can be used get the system up and running in non-production environments.

This document describes how ALES uses SSL and provides instructions for replacing the demonstration certificates with those signed by a recognized Certificate Authority. It contains the following topics:

ALES Component Connections

ALES uses one-way or two-way SSL as follows:

- **Administration Server and Remote ALES Components**

  Once the enrollment process is performed on a remote machine, all ALES components on that machine (SCM, SSM) are bound into the internal ALES trust structure based on the internal CA, residing on the Administration Server. All communication with the server is performed using two-way SSL.

- **SSM Enrollment Clients**

  With the exception of Web Services SSMs, a single set of keys located in *BEA_HOME*\ales30-shared\keys is used by all ALES components on that machine. When enrollment is initiated on a remote machine, communication between the enrollment client and the Administration Server is one-way SSL.

  NOTE: For step-by-step enrollment instructions, see the SSM Installation and Configuration Guide.

  If enrollment is performed in demo mode, the Administration Server presents its certificate signed by the Demo ALES CA that is supplied with the installation that enrollment clients are configured to trust. In secure mode, the client verifies the CA certificate against its list of trusted certificate authorities in $JAVA_HOME/lib/security/cacerts.

- **Internet Explorer Browsers**

  One-way SSL is used for browser connections with the Administration Console or the Entitlements Management Tool. When a browser client initiates the connection, the Administration Server sends the client its certificate. If the CA authority that signed the certificate of Administration web server (WebLogic or Tomcat) is in the browser's trusted stored, the browser proceeds to establish the one-way SSL connection. If not, the browser issues a warning that allows the user to trust the certificate.

NOTE: The ALES administration tools themselves use two-way SSL when communicating with other internal ALES components.

- **External Business Logic Manager (BLM) Clients**

  Instead of using the provided Java wrapper for the BLM SOAP interface, external clients may directly access BLM interfaces. For instructions, see

Demo Keystores and Certificates

Upon installation, two keystores containing demo certificates are used to establish trust between the Administration Server and clients:

– **`Webserver.jks`**—The Administration Server uses `BEA_HOME\ales30-shared\keys\webserver.jks`. This keystore contains a demonstration private key for the Administration Server, the server's identity in a public certificate that is signed by the Demo ALES CA, and a public certificate for the internal CA itself.

– **`DemoTrust.jks`**— SSM enrollment clients use this keystore when enrolling in `demo` mode. Because this keystore also contains the Demo CA certificate, clients will trust the Administration Server. This keystore is located in the `BEA_HOME\ales30-shared\keys` directory.

Replacing the Demo Certificates

For production environments, first configure the Administration Server's keystore to use a keystore containing a valid CA certificate. After this, SSMs can be bound into the SSL framework by enrolling in `secure` mode.

Note: Some certificates issued by CA authorities do not strictly comply with Certicom's Internet X.509 Public Key Infrastructure standard. To use these certificates, you must disable constraints extension checking by adding information to the the enrollment and unenrollment scripts. For instructions, see "Disable Constraints Extension Checking" on page 2-6.

Clients enrolling in `secure` mode will verify the CA certificate against its list of trusted certificate authorities in `$JAVA_HOME/lib/security/cacerts`, which already contains most commercial CAs. If the certificate authority you are using is not in the list of trusted CAs, the CA's certificate must be imported into `cacerts`.

1. Rename `BEA_HOME\ales30-shared\keys\webserver.jks` to `demowebserver.jks` or a similar name.

   Note: This allows you to create the new keystore named `webserver.jks`. Doing so will minimize modifications that must be made to existing Administration Server config files.

2. Using the Keytool utility, enter:

   ```
   keytool -genkey -alias ales-webserver -keyalg RSA -keystore Webserver.jks
   ```

3. When prompted, enter the keystore password and other information about the certificate, (company, contact name, etc.).

4. When prompted for the key password, enter the same password used for the keystore itself. This can be accomplished by pressing ENTER.

5. Create a Certificate Signing Request (CSR) as shown below and submit it to the Certificate Authority:

   ```
   keytool -certreq -alias ales-webserver -keyalg RSA -file certreq.csr
   -keystore Webserver.jks
   ```

6. When you receive the signed certificate, download a chain certificate from the CA.
7. Import the chain certificate and new CA certificate into the keystore:

   ```
   keytool -import -alias AlesCA -keystore Webserver.jks -trustcacerts -file
   <chain_certificate_filename>
   ```

   ```
   keytool -import -alias ales-webserver -keystore Webserver.jks
   -trustcacerts -file <certificate_filename>
   ```

8. Copy the new `Webserver.jks` to the `BEA_HOME\ales30-shared\keys` directory.
9. Modify the server's configuration file as described in the table below.

Container Type	Instructions
WebLogic Server	In `BEA_HOME/asiDomain/config.xml`, replace the existing `<server-private-key-pass-phrase-encrypted>` value with the encrypted value of the keystore password used when new `webserver.jks` keystore was created (see step 3 on page 2-4). To encrypt the password, you may use the `encrypt` tool provided with WebLogic Server.
Tomcat	Modify `TOMCAT_HOME/config/server.xml` as follows: Add `keystorePass=<encrypted_keystore_password>` next to the `keystoreFile` attribute.

10. Restart the Administration Server.

Configuring BLM Clients for One-Way SSL

SSL connections between BLM clients and the BLM server are two-way SSL by default. You can change this to one-way SSL using the following steps:

1. Open `BEA_HOME/ales30-admin/config/WLESblm.properties` in an editor and add the following parameter to the bottom of the file:

   ```
   BLM.sslType=one-way
   ```

   Note: If you are using the default properties file, this is already entered as a commented line at the bottom of the file. Simply remove the comment symbol (#).

2. Restart the server using the following command:

```
BEA_HOME/ales30-admin/bin/WLESadmin.sh restart
```

This is all that is required if the BLM client is on the same machine and the server. You do not need to perform the remaining steps.

3. If the BLM client is on a separate machine, make a copy of `trust.jks` in the `BEA_HOME/ales30-shared/keys` directory and move the copy to an appropriate directory on the BLM client machine.
4. On the BLM client machine, add the following parameter to the BLM client application:

   ```
   -Dwles.ssl.trustedCAKeyStore=/<directory_name>/trust.jks
   ```

   where

   <`directory_name`>—name of the directory containing `trust.jks`.

   Note: No keys are distributed with `trust.jks`. It contains only the CA public certificate.

Disable Constraints Extension Checking

If your CA certificates do not strictly comply with Certicom's Internet X.509 Public Key Infrastructure standard, you must disable constraints extension checking by the enrollment and unenrollment scripts. To do this, add the following lines to `enroll.bat|sh` and `unenroll.bat|sh` located in the `BEA_HOME/ales32-shared/bin` directory.

```
if [ -f $JAVA_HOME/lib/security/cacerts ]; then

JAVA_OPTIONS="-Dbea.home=$BEA_HOME -Dwles.ssl.enforceConstraints=false
-Dwles.ssl.verifyHostnames=yes
-Dwles.ssl.trustedCAKeyStore=$JAVA_HOME/lib/security/cacerts
-Dlog4j.configuration=file:./log4j.properties"

else

JAVA_OPTIONS="-Dbea.home=$BEA_HOME -Dwles.ssl.enforceConstraints=false
-Dwles.ssl.verifyHostnames=yes
-Dwles.ssl.trustedCAKeyStore=$JAVA_HOME/jre/lib/security/cacerts
-Dlog4j.configuration=file:./log4j.properties"

fileif [ "$1" = "demo" ]; then

JAVA_OPTIONS="-Dbea.home=$BEA_HOME -Dwles.ssl.enforceConstraints=false
-Dwles.ssl.verifyHostnames=no
-Dwles.ssl.trustedCAKeyStore=$ALES_SHARED_HOME/keys/DemoTrust.jks
-Dlog4j.configuration=file:./log4j.properties"
```

```
else
```

CHAPTER 3

Failover and System Reliability

This section describes ALES features that support recovery from failure. It contains the following topics:

Understanding Failover

In general, failover is the ability of a product to detect the failure of a particular component and switch to a working replica of that component without losing functionality. ALES support two failover scenarios:

- Runtime failover — makes sure that SSMs continue to provide security services even if the external components it relies on (such as the authentication database) become unavailable during runtime. This failover mechanism is achieved by configuring secondary sources of information for ALES security providers. See Figure 3-1 for an illustration of failover during runtime of an SSM.

- Administration time failover — makes sure that ALES administration services are accessible even if the primary Administration Server fails. This failover is achieved by configuring a secondary Administration Server. The secondary server is a redundant server that should be accessed if the primary one cannot be used. Figure 3-2 and Figure 3-3 show how ALES supports administration failover using a secondary Administration Server.

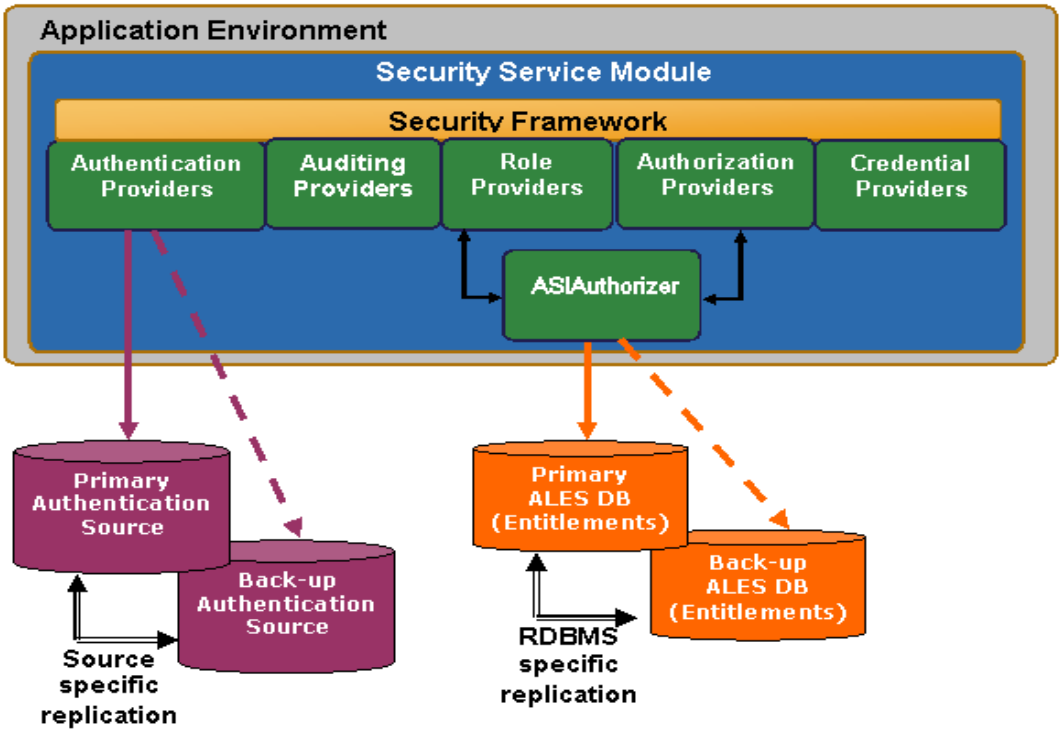
Assuring Runtime Failover for SSMs

ALES security providers depend on data stores for authentication, authorization, and credential mapping. You can configure ALES for failover in these important cases:

- Authentication failover is provided by configuring the SSM to point to primary and secondary user data stores. Replication of the data stores is handled by the native functionality of the data store, such as:
  - database replication for a relational database system
  - LDAP master/slave configuration
  - primary and secondary domain controllers in a Windows NT domain
- Credential mapping failover is provided by configuring the ALES Database Credential Mapper to use primary and secondary databases.

Note that SSMs have no runtime dependency on the Administration Server.

Figure 3-1 Runtime Availability

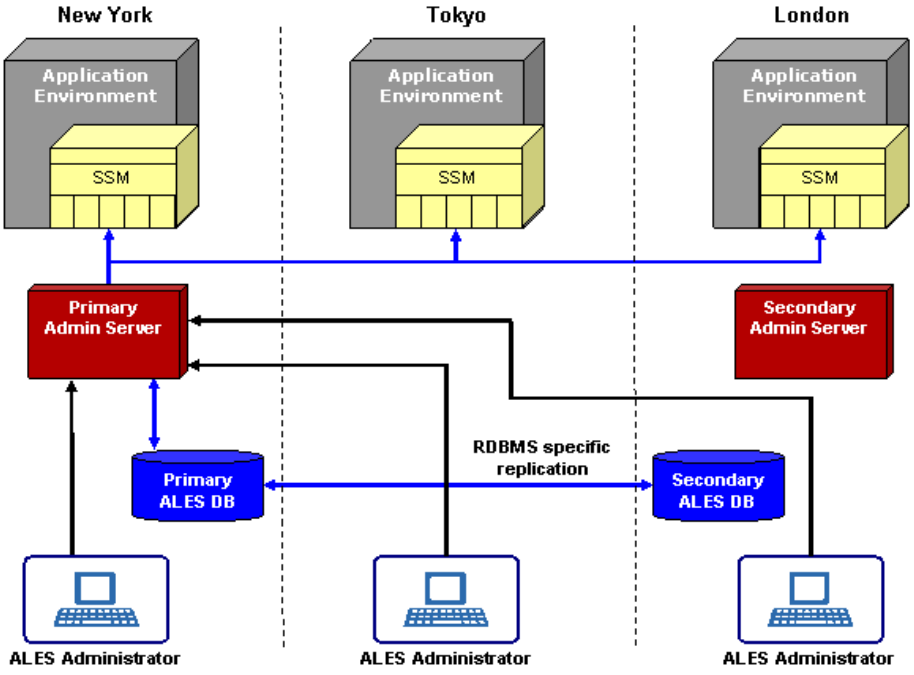


Assuring Administrative Server Availability

Failover for ALES administration functions can be achieved by installing a secondary, redundant Administration Server that will be used only when the primary becomes unavailable.

For example, consider the global deployment illustrated in Figure 3-2. The depicted enterprise has applications staged on servers in New York, Tokyo, and London. It has also deployed redundant ALES Administration Servers in its New York and London data centers and provides a replicated database to store ALES policies and entitlements information. Under normal conditions, administrators interact with the primary Administration Server in New York only. When policies are updated, the primary Administration Server pushes the changes to all SSMs in the global environment.

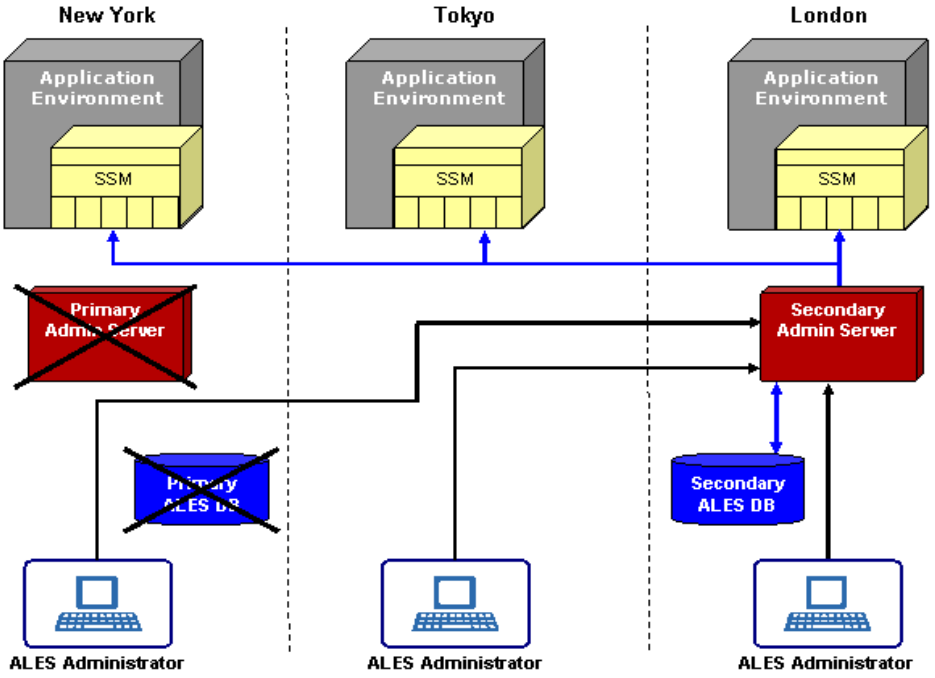
Figure 3-2 Administrative Availability (Working Normally)



If the data center in New York goes down, as illustrated in Figure 3-3, the SSMs detect this failure and connect to the secondary Administration Server. The secondary Administration Server uses the primary database unless it becomes unavailable, in which case the server connects to the secondary database server (the replica).

Note that both the primary and secondary Administration Server can use either the primary or secondary database servers.

Figure 3-3 Administrative Availability (After Failure)



One benefit of the ALES architecture is that even if all Administration Servers, including secondary Administration Servers go down (for maintenance or due to failure), there is no impact on the applications in production or on the security services provided by the SSMs.

For information on how to configure the Administration Server for failover, see "Setting up a Failover Administration Server" on page 3-9.

Failover Considerations for the Database Server

Figure 3-3 provides a logical view of failover functionality when the primary database server fails.

Because the database server contains all the configuration and security data used by the Administration Application to protect applications and resources, it must be highly available and

reliable. This can be accomplished by implementing the recommendations of the database manufacturer (for example, through the use of clustering architecture or hot standby).

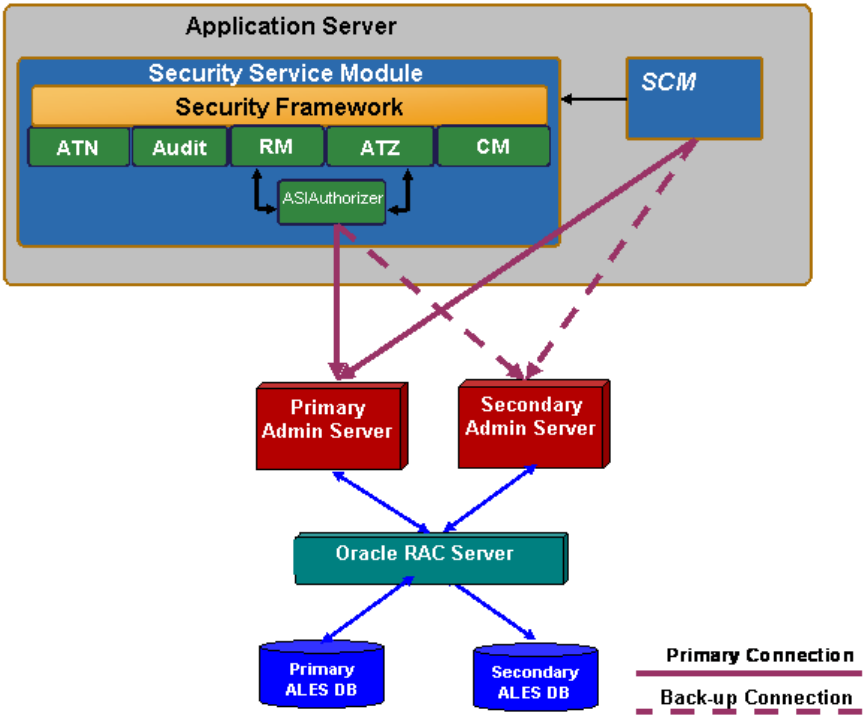
The number of redundant database servers you configure can vary; however, a minimum of two is recommended to maintain reliable services. It is up to the system administrator to set up database failover and configure data replication between the database instances.

There are two approaches for making sure that two instances of the ALES database contain the same data:

- Use Oracle RAC for Oracle databases (see Figure 3-4) or implement a similar approach recommended by the database vendor. This allows ALES providers to be configured with only one address, assuming transparent failover for the database is provided by the database vendor.
- Use the replication mechanism recommended by the database vendor. In this case, set up the primary database and secondary database with unique connection information. The connection information for the secondary database can be added to the ALES Database provider and the ALES Database Credential Mapper. You configure this connection information in the Administration Console on each provider's **Failover** tab.

Figure 3-4 illustrates the failover mechanism for the ALES Administration Server using Oracle RAC.

Figure 3-4 ALES Administration Servers using Oracle RAC



Common methods of achieving high availability include performing periodic back-ups, using fault tolerant disks, and manually copying files whenever they are changed. This is also the case for any optional external data sources in use. A database backup can be used for database recovery in the case of disk failure.

Failover Considerations for SSMs

Use the Administration Console to configure failover support for database-related and LDAP authentication providers by specifying the secondary database or LDAP server, as described in the following sections.

Secondary Databases

A secondary database can be configured for Database Credential Mapping, Database Authentication, ASI Authorization, and ASI Role Mapper providers.

The ASI Authorization Provider contacts an external process to evaluate its authorization queries. If that process dies, the ASI Authorization provider denies access to all resources. This provider can be configured to contact the Administration database to retrieve subject attributes and group membership for use in authorization and delegation decisions. If the database connection fails, the provider connects to the configured secondary database and will also try to reconnect to the failed database after a configurable time-out. If all database connections fail and defined policies operate on user attributes and group membership, all access is denied.

Secondary LDAP Servers

A secondary LDAP server can be configured for Novell LDAP, Active Directory, iPlanet, and Open LDAP authenticators.

The NT Authenticator already supports multiple domain controllers. The WebLogic Authenticator, WebLogic Authorizer and WebLogic Role Mapper use WebLogic's internal LDAP server as its data store. No support for a redundant source is required.

Failover Considerations for SCMs

When a SCM starts, it contacts the Administration Server to obtain and cache the most current configuration data. When configuration data is modified, the Administration Server pushes the updates to the SCM.

Failover for the SCM server is implemented as follows:

- An SCM can configured with addresses for primary and secondary Administration Servers. This can be specified during installation when the install program prompts for this information or specified after installation by modifying the following properties in the SCM's `SCM.properties` file:

  `domain.asi.primary.pdurl` — primary server address
  `domain.asi.secondary.pdurl` — secondary server address

  Note: If a secondary server is not specified during installation, both properties will specify the primary server.

- If no Administration Server is available, the SCM continues to operate using the previously cached policies and configuration data. If the SCM is starting for the first time or does not have a cache, it will continue searching for an Administration Server. Once a primary or

secondary Administration Server is available, the SCM connect to it and obtains its configuration data.

Setting up a Failover Administration Server

The following tasks must be performed to set up a secondary Administration Server to provide failover support:

1. Install the secondary Administration Server on a separate machine as described in "Install a Secondary Administration Server" on page 3-9.

   Note: For complete instructions on installing Administration Servers, see the Administration Server Installation Guide.

2. Initialize the secondary Administration Server's trust stores as described in "Initialize the Secondary Server Trust Stores" on page 3-10.

3. Enable periodic trust synchronization on the secondary Administration Server as described in "Enable Trust Synchronization" on page 3-11.

Install a Secondary Administration Server

The secondary Administration Server must installed on a separate machine and set up in the same manner as the primary Administration Server.

1. Install the container to host Administration Server (WebLogic Server or Apache Tomcat).

2. Run the Administration Server installation program to install the secondary Administration Server. When prompted, provide the information described in Table 3-1.

Table 3-1 Installing the Secondary Administration Server

Prompt	Description
Secondary Server URL	Leave blank

Table 3-1 Installing the Secondary Administration Server

Prompt	Description
Database Configuration	With the exception of the **Install Database Schema** property, specify the same information specified when the primary Administration Server was installed. **CAUTION:** Be sure to clear the **Install Database Schema** checkbox so that the schema is not installed.
Key Protection Password Selection	It is recommended that you select the **Advanced Password Configuration** option and then specify the passwords when prompted. The password entered do not have to match those on the primary server. If the **Advanced Password Configuration** option is used, the passwords can be used to decrypt SCM and SSM cache files, which may be useful for debugging.

When complete, initialize the secondary trust stores as described in the next section.

Initialize the Secondary Server Trust Stores

When the secondary Administration Server is installed, a set of unique certificates is generated for it. Before starting the server, you must synchronize its trust stores with those in use on the primary Administration Server.

To initialize the secondary Administration Server trust stores:

1. On the secondary Administration Server, create the following directories:

   `ALES_HOME/primary-shared-keys`

2. Copy the contents of the `ALES_SHARED/keys` directory from the primary Administration Server to the secondary Administration Server machine into the `ALES_HOME/primary-shared-keys` directory.

3. From the secondary server's `/bin` directory, execute `initialize_backup_trust.bat|.sh`. When prompted for the primary shared SSL directory, enter the path to the `ALES_HOME/primary-shared-keys` directory.

When complete, enable trust synchronization as described in the next section.

Enable Trust Synchronization

Whenever a new SSM/SCM is enrolled — or an existing SSM/SCM is unenrolled — only the trust stores on the primary Administration Server are updated. Unless the secondary Administration Server's trust store is also updated, problems may occur during failover, because the secondary Administration Server will not trust the new SSMs.

To insure that the trust stores on secondary Administration Server's are current, the secondary server can be configured to perform periodic synchronization with the primary server's trust stores.

Note: It is very important that the trust synchronization be enabled on the secondary Administration Server only.

To configure the trust synchronization on the secondary Administration Server:

1. Start the secondary Administration Server and access the Administration Console.
2. In the Administration Console, select **Administration Console** at the top of the navigation tree in the left pane.
3. In the right pane, select the **Failover** tab.
4. Select the **Backup** radio button. Then complete the fields as described in Figure 3-2 and click **Apply**.

Table 3-2 Secondary Server's Failover Tab

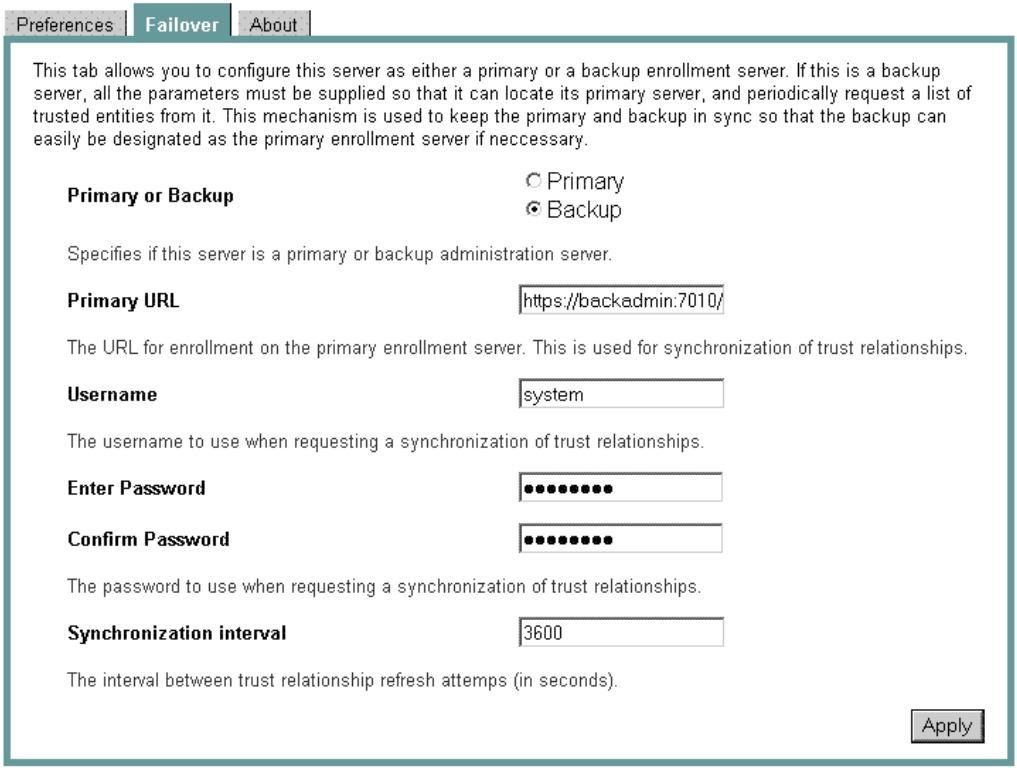
Field	Description
Primary URL	The URL of the primary Administration Server in the format: `https://<server_name>:7010/asi` where `<server_name>` is the server name or IP address. **Note:** This the same URL used to access the primary server's Administration Console.
Username	Name of ALES administrator user (the default is *system*).

Table 3-2 Secondary Server's Failover Tab

Field	Description
Enter Password & Confirm Password	Password of the ALES administrator user (the default is *weblogic*).
Synchronization Interval	Number of seconds between synchronization attempts This value depends on how frequently SSM or SCM instances are enrolled and un-enrolled with the primary Administration Server.

A sample completion of the **Failover** tab is shown in Figure 3-5.

Figure 3-5 Configuring a Backup Server in the Administration Console



CHAPTER 4

Performance Statistics

This section describes the ALES performance statistics feature, which enables collection of data about authentication and authorization for purposes of troubleshooting and performance analysis. It covers the following topics:

Enabling Performance Statistics Collection

The ALES performance statistic feature is controlled by an Auditing security provider, the PerfDBAuditor provider. Performance statistics are gathered for each Security Service Module in your ALES installation. In order to collect performance statistics for an SSM, you must enable and configure a PerfDBAuditor provider for that SSM.

Adding a PerfDBAuditor Provider

To add a PerfDBAuditor provider to an SSM other than a WebLogic Server SSM, use the ALES Administration Console. See "Using Performance Statistics with WebLogic Server 9.x\10.0" on page 4-2 for information about how to enable performance statistics collection with the WebLogic Server SSM.

1. Open the **Security Configuration** folder.

2. Open the **Service Control Manager** folder that contains the Security Service Module for which you want to enable performance statistics collection and then open the Security Service Module folder.
3. Open the **Auditing** folder, and click **Auditor**.
4. Click **Configure a new Perf DBAuditor**.
5. On the **General** tab, assign a name for the provider and click **Create**.
6. Click the **Details** tab and configure the PerfDBAuditor. See "Configuring Performance Statistics Collection" on page 4-3 for information about how to set these values.
7. Click **Apply**.

Note: Changes made to a provider do not take effect until after it is explicitly deployed and the associated Security Service Module is restarted.

After you have added a PerfDBAuditor provider to your SSM configuration, you can disable it either by removing it, or by clearing the **Enable Performance Statistics** checkbox on the provider's **Details** configuration page in the ALES Administration Console.

Using Performance Statistics with WebLogic Server 9.x\10.0

To add a PerfDBAuditor provider to a WebLogic Server SSM, use the WebLogic Server administration console:

1. In the WebLogic Server Administration Console, navigate to **Security Realms** > `<active security realm>` > **Providers** > **Auditing** and click **New**.

   The Create a New Auditing Provider page appears.
2. In the **Name** field, enter a name for the Auditing provider.
3. From the **Type** dropdown field, select **PerfDBAuditor** as the provider type and click **OK**.
4. Select **Providers** > **Auditing** and click the name of the new Auditing provider to complete its configuration.
5. On the Configuration: Provider-Specific page for the Auditing provider, set the desired values. See "Configuring Performance Statistics Collection" on page 4-3 for information about how to set these values.
6. Click **Save** to save your changes.

7. In the Change Center, click **Activate Changes** and then restart WebLogic Server.

After you have added a PerfDBAuditor provider to your SSM configuration, you can disable it either by removing it, or by clearing the **Enable Performance Statistics** checkbox on the provider's Provider-Specific configuration page in the WebLogic Server Administration Console. You must then restart WebLogic Server for this change to take effect.

Limitations of Performance Statistics in the WebLogic Server SSM

Performance statistics for authorization in the WebLogic Server SSM are available only if you use the ASI Authorization provider. Performance statistics for authentication in the WebLogic Server SSM are not available unless you use the SSM Java API for authentication.

Configuring Performance Statistics Collection

Any changes in the configuration of the PerfDBAuditor provider require restarting the SSM to take effect. You can configure the following settings in the PerfDBAuditor provider:

Basic Behavioral Settings

Performance Statistics Interval

The interval setting specifies data collection interval, in minutes. This determines the length of periods during which the performance statistics data is accumulated before it is dumped to the database tables. All of the internal statistics counters are reset at the beginning of each interval. It should be a positive integer number. Required. The default is 5 minutes.

Performance Statistics Duration

You can collect performance statistics either in circular buffer mode or continuous mode. Circular buffer mode means that, after a specified amount of time elapses, new records are written over the oldest records from the same SSM in the database tables. This prevents performance statistics from growing to an unlimited extent. In continuous mode, records are not overwritten, but there is no limit imposed by the performance statistics feature to the potential size of the database tables.

The Performance Statistics Duration setting specifies whether to operate in circular buffer mode or continuous mode. A positive integer value causes performance statistics to be collected in circular buffer mode and specifies, in minutes, how long the statistics collection proceeds before new records start to overwrite the oldest ones. A special value of 0 means that no loopback will occur; statistics collection proceeds in continuous mode. The value of this field should be either

a positive integer number, greater than the interval, or 0, which is the default. It is a required setting.

In either mode, when an SSM is restarted, all previously existing data is cleaned from the database. Performance statistics data is not preserved across SSM restarts.

Enable Performance Statistics

The Enable Performance Statistics checkbox specifies whether the performance statistics collection is enabled or disabled. It serves as a temporary means of disabling the statistics collection without removing the PerfDBAuditor provider from the SSM's configuration. You must restart the SSM after changing this setting before it will take effect. Required. The default is enabled.

Database Connection Settings

JDBC Driver Classname

Specifies which Java class will be used for communication with the database. Required; the default is `oracle.jdbc.driver.OracleDriver`.

JDBC Connection URL

Specifies the connection string to use with the specified driver class. Formats for the database URL and driver class name vary depending on the type of database you are using. For example:

- `jdbc:oracle:thin:@<hostname>:<portnum>:SID` or
- `jdbc:sybase:Tds:<hostname>:<portnum>/<dbname>`

Required.

Database User Login

Specifies the login name of database user with sufficient rights for working with the performance-related tables. This user must possess write and delete privileges for those tables. Required.

Database User Password

The password for the database user specified in the login setting. This password will be stored, in an encrypted form, in the ALES User Store and distributed to the SSM for accessing the database. Required.

JDBC Connection Properties

A parameter for specifying any additional database connection properties that may be needed, in name=value format. Optional.

Database Table Settings

The following specify elements of the database schema used for storing performance statistics data. The default database tables are part of the default ALES database schema. If you for some reason need to use different tables, you need to create them yourself in your database schema.

Authentication Statistics Table

The name of the table that contains authentication-related performance statistics. Optional, but at least one of Authentication Statistics Table or Authorization Statistics Table must be present. Default value is PERF_ATH_STAT.

Authorization Statistics Table

The name of the table that contains authorization-related performance statistics. Optional, but at least one of Authentication Statistics Table or Authorization Statistics Table must be present. Default value is PERF_ATZ_STAT.

Authorization Attributes Statistics Table

The name of the table that contains authorization attributes-related performance statistics. Optional. The default value is PERF_ATZ_ATTR_STAT.

Authorization Functions Statistics Table

The name of the table that contains authorization functions-related performance statistics. Optional. The default value is PERF_ATZ_FUNC_STAT.

Using Performance Statistics

The ALES performance statistics feature gathers the following information, for each SSM configuration ID and host name, aggregated for each time interval specified by the Performance Statistics Interval setting:

- Number of requested and successful authentications
- Number of requested and successful authorizations

- Average latency of an authentication request, in milliseconds
- Average latency of an authorization request (the duration of calls to `isAccessAllowed` from start to end), in milliseconds
- For any user attribute required for policy evaluation or response:
  - Average retrieval time, in milliseconds
  - Total number of retrievals
- For each external function called during evaluation:
  - Average execution time, in milliseconds
  - Total number of calls

Performance statistics are stored in the database tables described in "Performance Statistics Database Schema" on page 4-6. To access the performance statistics, use SQL to retrieve the information you are interested in.

When an SSM is restarted, all previously existing data is cleaned from the database. Performance statistics data is not preserved across SSM restarts. Note also that performance statistics entries are uniquely identified by hostname and the configuration ID of the SSM. If you have two SSMs on the same host with the same configuration ID, their performance records will collide and only one will be stored successfully.

Performance Statistics Database Schema

Performance statistics are stored in four tables in the standard ALES database schema:

Authentication Statistics Table: PERF_ATH_STAT

This table contains authorization-related performance statistics.

Table 4-1 Authentication Statistics Table: PERF_ATH_STAT

Column	Type	Description
location	varchar(100)	The SSM that is the source of the statistics, recorded as *<hostname>* + *<SSM Configuration ID>* + `AthEvent`
id	number(12)	A sequential record ID.
starttime	date	The starting time of the interval.

Table 4-1 Authentication Statistics Table: PERF_ATH_STAT

Column	Type	Description
interval	number(12)	The length of the interval in seconds.
totalreq	number(12)	The total number of authentication requests during the interval.
successes	number(12)	The number of successful authentication requests during the interval.
avrlatency	float(10)	Average request latency in milliseconds.

Authorization Statistics Table: PERF_ATZ_STAT

This table contains authorization-related performance statistics.

Table 4-2 Authorization Statistics Table: PERF_ATZ_STAT

Column	Type	Description
location	varchar(100)	The SSM that is the source of the statistics, recorded as <*hostname*> + <*SSM Configuration ID*> + `AtzEvent`
id	number(12)	A sequential record ID.
starttime	date	The starting time of the interval.
interval	number(12)	The length of the interval in seconds.
totalreq	number(12)	The total number of authorization requests during the interval.
successes	number(12)	The number of successful authorization requests during the interval.
avrlatency	float(10)	Average request latency in milliseconds.

Authorization Attributes Statistics Table: PERF_ATZ_ATTR_STAT

This table contains performance statistics related to user attributes required for policy evaluation during authorization.

Table 4-3 Authorization Attributes Statistics Table: PERF_ATZ_ATTR_STAT

Column	Type	Description
location	varchar(100)	The SSM that is the source of the statistics, recorded as <*hostname*> + <*SSM Configuration ID*>> + AtzAttr
id	number(12)	A sequential record ID.
name	varchar(100)	The name of the attribute for which statistics are collected.
totalreq	number(12)	The total number of authorization requests requiring this user attribute for evaluation during the interval.
avrlatency	float(10)	Average request latency in milliseconds.

Authorization Functions Statistics Table: PERF_ATZ_FUNC_STAT

This table contains performance statistics related to external functions called during authorization.

Table 4-4 Authorization Functions Statistics Table: PERF_ATZ_FUNC_STAT

Column	Type	Description
location	varchar(100)	The SSM that is the source of the statistics, recorded as <*hostname*> + <*SSM Configuration ID*> + AtzAttr
id	number(12)	A sequential record ID.
name	varchar(100)	The name of the external function for which statistics are collected.
totalreq	number(12)	The total number of authorization requests calling this external function during the interval.
avrlatency	float(10)	Average request latency in milliseconds.

CHAPTER 5

Host Name or IP Address Change

This section describes how to reconfigure ALES after the initial installation. The following topics are described:

Configuration for New Host Name

This section describes how to reconfigure components if you change the host name of the system on which the Administration server is installed. The steps you follow depend on whether you are enrolling the SSM instance in demo or secure mode:

Admin Server and SSM on Same WLS System, SSM Instance in Demo Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is WebLogic Server.

Follow these steps to reconfigure ALES:

1. Shut down all ALES services, including the ALES Administration Server and the SCM:
   a. *BEA_HOME*/ales30-admin/bin/WLESadmin.sh stop
   b. *BEA_HOME*/ales30-scm/bin/WLESscm.sh stop
   c. Stop the SSM instance.

   If any of these components was started in console mode, type CTRL-C to stop it.
2. Modify *BEA_HOME*/ales30-scm/apps/scm-asi/sar-inf/config.xml to replace the old host name with the new host name:
   a. <listener host="*<OLD HOSTNAME>*" port="7013" protocol="https">
   a. <proxy best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   b. <pd best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   c.)<scm domain="asi" localName="adminconfig", instanceName="SCM.*<OLD HOSTNAME>*.asi"/>
3. Modify *BEA_HOME*/ales30-admin/config/WLESadmin.bat to replace the old host name with the new host name:
   a. Change https://*<OLD-HOSTNAME>*:7011/services/ManagedServer
   b. Change https://*<OLD-HOSTNAME>*:7013 to https://*<NEW-HOSTNAME>*:7013.
   c. Change https://*<OLD-HOSTNAME>*:7013
4. Modify *BEA_HOME*/ales30-admin/config/WLESWebLogic.conf to replace the old host name with the new host name:

a. Change `https://<OLD-HOSTNAME>` to `https://<NEW-HOSTNAME>`.

5. Modify *BEA_HOME*`/ales30-admin/config/WLESarme.properties` to replace the old host name with the new host name:

   a. Change `PDAddress = https://<OLD-HOSTNAME>` to `PDADDRESS = https://<NEW-HOSTNAME>`.

   b. Change `instanceName = ARME.admin.server.asi.<OLD-HOSTNAME>` to `instanceName = ARME.admin.server.asi.<NEW-HOSTNAME>`.

6. Modify *BEA_HOME*`/ales30-admin/config/WLESblm.properties` to replace the old host name with the new host name:

   a. Change `BLM.host = <OLD-HOSTNAME>` to `BLM.host = <NEW-HOSTNAME>.`

7. Modify *BEA_HOME*`/ales30-admin/config/asi.properties` to replace the old host name with the new host name:

   a. Change `ASI.BLMAddresses` to replace the old host name with the new host name.

   b. Change `ASI.ARMEAddresses` to replace the old host name with the new host name.

   c. Change `ASI.PDAddresses` to replace the old host name with the new host name.

8. Modify *BEA_HOME*`/ales30-ssm/<ssm>/adm/ssm_install.properties` to replace the old host name with the new host name:

   a. Change `host.name = <OLD-HOSTNAME>` to `host.name = <NEW-HOSTNAME>`.

   b. Change `scm.primary.admin.url = https://<OLD-HOSTNAME>:7010` to `scm.primary.admin.url = https://<NEW-HOSTNAME>:7010`.

   c. Change `scm.backup.admin.url = https://<OLD-HOSTNAME>:7010` to `scm.backup.admin.url = https://<NEW-HOSTNAME>:7010`.

   d. If the host name of the Admin server is changed, change `admin.host`.

9. If the host name of the SSM is changed, modify *BEA_HOME*`/ales30-ssm/<ssm>/adm/silient_install_ssm.xml` as follows:

   a. Change `<data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<old ssm host name>:7010/" />`

   b. Change `<data-value name="SCM_BACKUP_ADMIN_URL" value="https://<old ssm host name>:7010/" />`

10. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/adm/ssm_instance.properties to replace the old host name with the new host name:

    a. Change host.name = <*OLD-HOSTNAME*> to host.name = <*NEW-HOSTNAME*>.

    b. If the host name of the ALES Administration host is also changed, then change admin.host = <*OLD-HOSTNAME*> to admin.host = <*NEW-HOSTNAME*>.

11. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/bin/set-wls-env.bat to replace the old host name with the new host name:

    a. Change -Dwles.config.signer=<*OLD-HOSTNAME*> to -Dwles.config.signer=<*NEW-HOSTNAME*>.

12. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/config/WLESWeblogic.conf or WLESws.wrapper.conf to replace the old host name with the new host name.

13. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/config/WLESarme.properties to replace the old host name with the new host name:

    a. Change PDAddress = https://<*OLD-HOSTNAME*> to PDADDRESS = https://<*NEW-HOSTNAME*>.

    b. Change instanceName = ARME.....asi.<OLD-HOSTNAME> to instanceName = ARME....asi.<*NEW-HOSTNAME*>.

14. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of wles, a password of password, and a sybase server name of MYHOST, perform the following steps:

    a. Start one command window and run isql -Dwles -Ppassword -SMYHOST.

    b. Run select * from engine_addresses to verify the old address are being used.

    c. Delete the records in the engnie_addresses table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. sqlplus wles/password@ASI

    b. select * from engine_address;

    c. truncate table engine_addresses;

Admin Server and SSM on Same WLS System, SSM Instance in Secure Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is WebLogic Server.

Follow these steps to reconfigure ALES:

1. Run `unenroll.bat secure` **before** you change the host name.
2. Shut down all ALES services, including the ALES Administration Server and the SCM:
   a. *BEA_HOME*`/ales30-admin/bin/WLESadmin.sh stop`
   b. *BEA_HOME*`/ales30-scm/bin/WLESscm.sh stop`
   c. Stop the SSM instance.

   If any of these components was started in console mode, type CTRL-C to stop it.
3. Modify *BEA_HOME*`/ales30-scm/apps/scm-asi/sar-inf/config.xml` to replace the old host name with the new host name:
   a. `<listener host="<`*OLD HOSTNAME*`>" port="7013" protocol="https">`
   a. `<proxy best="<`*OLD HOSTNAME*`>" port="7011" protocol="https">`
   b. `<pd best="<`*OLD HOSTNAME*`>" port="7011" protocol="https">`
   c. `)<scm domain="asi" localName="adminconfig", instanceName="SCM.<`*OLD HOSTNAME*`>.asi"/>`
4. Modify *BEA_HOME*`/ales30-admin/bin/WLESadmin.bat` to replace the old host name with the new host name:
   a. Change `https://<`*OLD-HOSTNAME*`>:7011/services/ManagedServer`
   b. Change `https://<`*OLD-HOSTNAME*`>:7013` to `https://<`*NEW-HOSTNAME*`>:7013`.
   c. Change `https://<`*OLD-HOSTNAME*`>:7013`
5. Modify *BEA_HOME*`/ales30-admin/config/WLESWebLogic.conf` to replace the old host name with the new host name:
   a. Change `https://<`*OLD-HOSTNAME*`>` to `https://<`*NEW-HOSTNAME*`>`.
6. Modify *BEA_HOME*`/ales30-admin/config/WLESarme.properties` to replace the old host name with the new host name:

a. Change `PDAddress = https://<OLD-HOSTNAME>` to `PDADDRESS = https://<NEW-HOSTNAME>`.

b. Change `instanceName = ARME.admin.server.asi.<OLD-HOSTNAME>` to `instanceName = ARME.admin.server.asi.<NEW-HOSTNAME>`.

7. Modify *BEA_HOME*`/ales30-admin/config/WLESblm.properties` to replace the old host name with the new host name:

   a. Change `BLM.host = <OLD-HOSTNAME>` to `BLM.host = <NEW-HOSTNAME>.`

8. Modify *BEA_HOME*`/ales30-admin/config/asi.properties` to replace the old host name with the new host name:

   a. Change `ASI.BLMAddresses` to replace the old host name with the new host name.

   b. Change `ASI.ARMEAddresses` to replace the old host name with the new host name.

   c. Change `ASI.PDAddresses` to replace the old host name with the new host name.

9. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/ssm_install.properties` to replace the old host name with the new host name:

   a. Change `host.name = <OLD-HOSTNAME>` to `host.name = <NEW-HOSTNAME>`.

   b. Change `scm.primary.admin.url = https://<OLD-HOSTNAME>:7010` to `scm.primary.admin.url = https://<NEW-HOSTNAME>:7010`.

   c. Change `scm.backup.admin.url = https://<OLD-HOSTNAME>:7010` to `scm.backup.admin.url = https://<NEW-HOSTNAME>:7010`.

   d. If the host name of the Admin server is changed, change `admin.host`.

10. If the host name of the SSM is changed, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/silient_install_ssm.xml` as follows:

    a. Change `<data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<old ssm host name>:7010/" />`

    b. Change `<data-value name="SCM_BACKUP_ADMIN_URL" value="https://<old ssm host name>:7010/" />`

11. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/adm/ssm_instance.properties` to replace the old host name with the new host name:

    a. Change `host.name = <OLD-HOSTNAME>` to `host.name = <NEW-HOSTNAME>`.

b. If the host name of the ALES Administration host is also changed, then change `admin.host = <OLD-HOSTNAME>` to `admin.host = <NEW-HOSTNAME>`.

12. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/bin/set-wls-env.bat` to replace the old host name with the new host name:

    a. Change `-Dwles.config.signer=<`*OLD-HOSTNAME*`>` to `-Dwles.config.signer=<`*NEW-HOSTNAME*`>`.

13. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/config/WLESWeblogic.conf` or `WLESws.wrapper.conf` to replace the old host name with the new host name.

14. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/config/WLESarme.properties` to replace the old host name with the new host name:

    a. Change `PDAddress = https://<`*OLD-HOSTNAME*`>` to `PDADDRESS = https://<`*NEW-HOSTNAME*`>`.

    b. Change `instanceName = ARME.....asi.<OLD-HOSTNAME>` to `instanceName = ARME....asi.<`*NEW-HOSTNAME*`>`.

15. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of `wles`, a password of `password`, and a sybase server name of `MYHOST`, perform the following steps:

    a. Start one command window and run `isql -Dwles -Ppassword -SMYHOST`.

    b. Run `select * from engine_addresses` to verify the old address are being used.

    c. Delete the records in the `engnie_addresses` table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. `sqlplus wles/password@ASI`

    b. `select * from engine_address;`

    c. `truncate table engine_addresses;`

16. Run `enroll.bat secure` to enroll the SSM instance with the new host name.

Admin Server and SSM on Same Tomcat System, SSM Instance in Demo Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is Apache Tomcat.

Follow these steps to reconfigure ALES:

1. Shut down all ALES services, including the ALES Administration Server and the SCM:
   a. *BEA_HOME*/ales30-admin/bin/WLESadmin.sh stop
   b. *BEA_HOME*/ales30-scm/bin/WLESscm.sh stop
   c. Stop the SSM instance.

   If any of these components was started in console mode, type CTRL-C to stop it.
2. Modify *BEA_HOME*/ales30-scm/apps/scm-asi/sar-inf/config.xml to replace the old host name with the new host name:
   a. <listener host="*<OLD HOSTNAME>*" port="7013" protocol="https">
   a. <proxy best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   b. <pd best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   c.)<scm domain="asi" localName="adminconfig", instanceName="SCM.*<OLD HOSTNAME>*.asi"/>
3. Modify *BEA_HOME*/ales30-admin/bin/WLESadmin.bat to replace the old host name with the new host name:
   a. Change https://*<OLD-HOSTNAME>*:7011/services/ManagedServer
   b. Change https://*<OLD-HOSTNAME>*:7013 to https://*<NEW-HOSTNAME>*:7013.
   c. Change https://*<OLD-HOSTNAME>*:7013
4. Modify *BEA_HOME*/ales30-admin/config/WLESTomcat.conf to replace the old host name with the new host name:
   a. Change https://*<OLD-HOSTNAME>* to https://*<NEW-HOSTNAME>*.
5. Modify *BEA_HOME*/ales30-admin/config/WLESarme.properties to replace the old host name with the new host name:
   a. Change PDAddress = https://*<OLD-HOSTNAME>* to PDADDRESS = https://*<NEW-HOSTNAME>*.
   b. Change instanceName = ARME.admin.server.asi.*<OLD-HOSTNAME>* to instanceName = ARME.admin.server.asi.*<NEW-HOSTNAME>*.
6. Modify *BEA_HOME*/ales30-admin/config/WLESblm.properties to replace the old host name with the new host name:

a. Change `BLM.host = <OLD-HOSTNAME>` to `BLM.host = <NEW-HOSTNAME>.`

7. Modify *BEA_HOME*`/ales30-admin/config/asi.properties` to replace the old host name with the new host name:

   a. Change `ASI.BLMAddresses` to replace the old host name with the new host name.

   b. Change `ASI.ARMEAddresses` to replace the old host name with the new host name.

   c. Change `ASI.PDAddresses` to replace the old host name with the new host name.

8. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/ssm_install.properties` to replace the old host name with the new host name:

   a. Change `host.name = <`*OLD-HOSTNAME*`>` to `host.name = <`*NEW-HOSTNAME*`>`.

   b. Change `scm.primary.admin.url = https://<`*OLD-HOSTNAME*`>:7010` to `scm.primary.admin.url = https://<`*NEW-HOSTNAME*`>:7010`.

   c. Change `scm.backup.admin.url = https://<`*OLD-HOSTNAME*`>:7010` to `scm.backup.admin.url = https://<`*NEW-HOSTNAME*`>:7010`.

   d. If the host name of the Admin server is changed, change `admin.host`.

9. If the host name of the SSM is changed, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/silent_install_ssm.xml` as follows:

   a. Change `<data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`

   b. Change `<data-value name="SCM_BACKUP_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`

10. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/adm/ssm_instance.properties` to replace the old host name with the new host name:

    a. Change `host.name = <`*OLD-HOSTNAME*`>` to `host.name = <`*NEW-HOSTNAME*`>`.

    b. If the host name of the ALES Administration host is also changed, then change `admin.host = <`*OLD-HOSTNAME*`>` to `admin.host = <`*NEW-HOSTNAME*`>`.

11. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/bin/set-wls-env.bat` to replace the old host name with the new host name:

a. Change `-Dwles.config.signer=<OLD-HOSTNAME>` to `-Dwles.config.signer=<NEW-HOSTNAME>`.

12. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/config/WLESTomcat.conf` or `WLESws.wrapper.conf` to replace the old host name with the new host name.

13. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/config/WLESarme.properties` to replace the old host name with the new host name:

    a. Change `PDAddress = https://<`*OLD-HOSTNAME*`>` to `PDADDRESS = https://<`*NEW-HOSTNAME*`>`.

    b. Change `instanceName = ARME.....asi.<OLD-HOSTNAME>` to `instanceName = ARME....asi.<`*NEW-HOSTNAME*`>`.

14. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of `wles`, a password of `password`, and a sybase server name of `MYHOST`, perform the following steps:

    a. Start one command window and run `isql -Dwles -Ppassword -SMYHOST`.

    b. Run `select * from engine_addresses` to verify the old address are being used.

    c. Delete the records in the `engnie_addresses` table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. `sqlplus wles/password@ASI`

    b. `select * from engine_address;`

    c. `truncate table engine_addresses;`

Admin Server and SSM on Same Tomcat System, SSM Instance in Secure Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is Apache Tomcat.

Follow these steps to reconfigure ALES:

1. Run `unenroll.bat secure` **before** you change the host name.

2. Shut down all ALES services, including the ALES Administration Server and the SCM:

    a. *BEA_HOME*`/ales30-admin/bin/WLESadmin.sh stop`

    b. *BEA_HOME*`/ales30-scm/bin/WLESscm.sh stop`

c. Stop the SSM instance.

If any of these components was started in console mode, type CTRL-C to stop it.

3. Modify *BEA_HOME*/ales30-scm/apps/scm-asi/sar-inf/config.xml to replace the old host name with the new host name:

   a. <listener host="*<OLD HOSTNAME>*" port="7013" protocol="https">

   a. <proxy best="*<OLD HOSTNAME>*" port="7011" protocol="https">

   b. <pd best="*<OLD HOSTNAME>*" port="7011" protocol="https">

   c.)<scm domain="asi" localName="adminconfig", instanceName="SCM.*<OLD HOSTNAME>*.asi"/>

4. Modify *BEA_HOME*/ales30-admin/bin/WLESadmin.bat to replace the old host name with the new host name:

   a. Change https://*<OLD-HOSTNAME>*:7011/services/ManagedServer

   b. Change https://*<OLD-HOSTNAME>*:7013 to https://*<NEW-HOSTNAME>*:7013.

   c. Change https://*<OLD-HOSTNAME>*:7013

5. Modify *BEA_HOME*/ales30-admin/bin/WLESTomcat.conf to replace the old host name with the new host name:

   a. Change https://*<OLD-HOSTNAME>* to https://*<NEW-HOSTNAME>*.

6. Modify *BEA_HOME*/ales30-admin/config/WLESarme.properties to replace the old host name with the new host name:

   a. Change PDAddress = https://*<OLD-HOSTNAME>* to PDADDRESS = https://*<NEW-HOSTNAME>*.

   b. Change instanceName = ARME.admin.server.asi.*<OLD-HOSTNAME>* to instanceName = ARME.admin.server.asi.*<NEW-HOSTNAME>*.

7. Modify *BEA_HOME*/ales30-admin/config/WLESblm.properties to replace the old host name with the new host name:

   a. Change BLM.host = *<OLD-HOSTNAME>* to BLM.host = *<NEW-HOSTNAME>.*

8. Modify *BEA_HOME*/ales30-admin/config/asi.properties to replace the old host name with the new host name:

   a. Change ASI.BLMAddresses to replace the old host name with the new host name.

b. Change `ASI.ARMEAddresses` to replace the old host name with the new host name.

c. Change `ASI.PDAddresses` to replace the old host name with the new host name.

9. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/ssm_install.properties` to replace the old host name with the new host name:

   a. Change `host.name = <`*OLD-HOSTNAME*`>` to `host.name = <`*NEW-HOSTNAME*`>`.

   b. Change `scm.primary.admin.url = https://<`*OLD-HOSTNAME*`>:7010` to `scm.primary.admin.url = https://<`*NEW-HOSTNAME*`>:7010`.

   c. Change `scm.backup.admin.url = https://<`*OLD-HOSTNAME*`>:7010` to `scm.backup.admin.url = https://<`*NEW-HOSTNAME*`>:7010`.

   d. If the host name of the Admin server is changed, change `admin.host`.

10. If the host name of the SSM is changed, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/silient_install_ssm.xml` as follows:

    a. Change `<data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`

    b. Change `<data-value name="SCM_BACKUP_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`

11. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/adm/ssm_instance.properties` to replace the old host name with the new host name:

    a. Change `host.name = <`*OLD-HOSTNAME*`>` to `host.name = <`*NEW-HOSTNAME*`>`.

    b. If the host name of the ALES Administration host is also changed, then change `admin.host = <`*OLD-HOSTNAME*`>` to `admin.host = <`*NEW-HOSTNAME*`>`.

12. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/bin/set-wls-env.bat` to replace the old host name with the new host name:

    a. Change `-Dwles.config.signer=<`*OLD-HOSTNAME*`>` to `-Dwles.config.signer=<`*NEW-HOSTNAME*`>`.

13. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/config/WLESTomcat.conf` or `WLESws.wrapper.conf` to replace the old host name with the new host name.

14. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/template/config/WLESarme.properties` to replace the old host name with the new host name:

a. Change `PDAddress = https://<OLD-HOSTNAME>` to `PDADDRESS = https://<NEW-HOSTNAME>`.

b. Change `instanceName = ARME.....asi.<OLD-HOSTNAME>` to `instanceName = ARME....asi.<NEW-HOSTNAME>`.

15. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of `wles`, a password of `password`, and a sybase server name of `MYHOST`, perform the following steps:

    a. Start one command window and run `isql -Dwles -Ppassword -SMYHOST`.

    b. Run `select * from engine_addresses` to verify the old address are being used.

    c. Delete the records in the `engnie_addresses` table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. `sqlplus wles/password@ASI`

    b. `select * from engine_address;`

    c. `truncate table engine_addresses;`

16. Run `enroll.bat secure` to enroll the SSM instance with the new host name.

Admin Server and SSM on Different System, Hostname of Admin Server Changed, SSM Instance in Demo Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is WebLogic Server.

Follow these steps to reconfigure ALES:

1. Shut down all ALES services, including the ALES Administration Server and the SCM:

   a. *BEA_HOME*`/ales30-admin/bin/WLESadmin.sh stop`

   b. *BEA_HOME*`/ales30-scm/bin/WLESscm.sh stop`

   c. Stop the SSM instance.

   If any of these components was started in console mode, type CTRL-C to stop it.

2. Modify *BEA_HOME*`/ales30-scm/apps/scm-asi/sar-inf/config.xml` to replace the old host name with the new host name:

   a. `<listener host="<OLD HOSTNAME>" port="7013" protocol="https">`

a. <proxy best="*<OLD HOSTNAME>*" port="7011" protocol="https">

b. <pd best="*<OLD HOSTNAME>*" port="7011" protocol="https">

c.)<scm domain="asi" localName="adminconfig", instanceName="SCM.*<OLD HOSTNAME>*.asi"/>

3. Modify *BEA_HOME*/ales30-admin/bin/WLESadmin.bat to replace the old host name with the new host name:

   a. Change https://*<OLD-HOSTNAME>*:7011/services/ManagedServer

   b. Change https://*<OLD-HOSTNAME>*:7013 to https://*<NEW-HOSTNAME>*:7013.

   c. Change https://*<OLD-HOSTNAME>*:7013

4. Modify *BEA_HOME*/ales30-admin/config/WLESWebLogic.conf to replace the old host name with the new host name:

   a. Change https://*<OLD-HOSTNAME>* to https://*<NEW-HOSTNAME>*.

5. Modify *BEA_HOME*/ales30-admin/config/WLESarme.properties to replace the old host name with the new host name:

   a. Change PDAddress = https://*<OLD-HOSTNAME>* to PDADDRESS = https://*<NEW-HOSTNAME>*.

   b. Change instanceName = ARME.admin.server.asi.*<OLD-HOSTNAME>* to instanceName = ARME.admin.server.asi.*<NEW-HOSTNAME>*.

6. Modify *BEA_HOME*/ales30-admin/config/WLESblm.properties to replace the old host name with the new host name:

   a. Change BLM.host = *<OLD-HOSTNAME>* to BLM.host = *<NEW-HOSTNAME>.*

7. Modify *BEA_HOME*/ales30-admin/config/asi.properties to replace the old host name with the new host name:

   a. Change ASI.BLMAddresses to replace the old host name with the new host name.

   b. Change ASI.ARMEAddresses to replace the old host name with the new host name.

   c. Change ASI.PDAddresses to replace the old host name with the new host name.

8. Modify *BEA_HOME*/ales30-ssm/*<ssm>*/adm/ssm_install.properties to replace the old host name with the new host name:

   a. If the host name of the Admin server is changed, change admin.host.

9. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/adm/ssm_instance.properties to replace the old host name with the new host name:

   a. If the host name of the ALES Administration host is also changed, then change admin.host = <*OLD-HOSTNAME*> to admin.host = <*NEW-HOSTNAME*>.

10. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/config/WLESarme.properties to replace the old host name with the new host name:

    a. Change PDAddress = https://<*OLD-HOSTNAME*> to PDADDRESS = https://<*NEW-HOSTNAME*>.

    b. Change instanceName = ARME.....asi.<OLD-HOSTNAME> to instanceName = ARME....asi.<*NEW-HOSTNAME*>.

11. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of wles, a password of password, and a sybase server name of MYHOST, perform the following steps:

    a. Start one command window and run isql -Dwles -Ppassword -SMYHOST.

    b. Run select * from engine_addresses to verify the old address are being used.

    c. Delete the records in the engnie_addresses table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. sqlplus wles/password@ASI

    b. select * from engine_address;

    c. truncate table engine_addresses;

Admin Server and SSM on Different System, Hostname of Admin Server Changed, SSM Instance in Secure Mode

In this scenario, the Admin server and the SSM are both installed on the same system, and the platform is WebLogic Server.

Follow these steps to reconfigure ALES:

1. Shut down all ALES services, including the ALES Administration Server and the SCM:

   a. *BEA_HOME*/ales30-admin/bin/WLESadmin.sh stop

   b. *BEA_HOME*/ales30-scm/bin/WLESscm.sh stop

   c. Stop the SSM instance.

If any of these components was started in console mode, type CTRL-C to stop it.

2. Modify *BEA_HOME*/ales30-scm/apps/scm-asi/sar-inf/config.xml to replace the old host name with the new host name:
   a. <listener host="*<OLD HOSTNAME>*" port="7013" protocol="https">
   a. <proxy best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   b. <pd best="*<OLD HOSTNAME>*" port="7011" protocol="https">
   c.)<scm domain="asi" localName="adminconfig", instanceName="SCM.*<OLD HOSTNAME>*.asi"/>
3. Modify *BEA_HOME*/ales30-admin/bin/WLESadmin.bat to replace the old host name with the new host name:
   a. Change https://*<OLD-HOSTNAME>*:7011/services/ManagedServer
   b. Change https://*<OLD-HOSTNAME>*:7013 to https://*<NEW-HOSTNAME>*:7013.
   c. Change https://*<OLD-HOSTNAME>*:7013
4. Modify *BEA_HOME*/ales30-admin/config/WLESWebLogic.conf to replace the old host name with the new host name:
   a. Change https://*<OLD-HOSTNAME>* to https://*<NEW-HOSTNAME>*.
5. Modify *BEA_HOME*/ales30-admin/config/WLESarme.properties to replace the old host name with the new host name:
   a. Change PDAddress = https://*<OLD-HOSTNAME>* to PDADDRESS = https://*<NEW-HOSTNAME>*.
   b. Change instanceName = ARME.admin.server.asi.*<OLD-HOSTNAME>* to instanceName = ARME.admin.server.asi.*<NEW-HOSTNAME>*.
6. Modify *BEA_HOME*/ales30-admin/config/WLESblm.properties to replace the old host name with the new host name:
   a. Change BLM.host = *<OLD-HOSTNAME>* to BLM.host = *<NEW-HOSTNAME>.*
7. Modify *BEA_HOME*/ales30-admin/config/asi.properties to replace the old host name with the new host name:
   a. Change ASI.BLMAddresses to replace the old host name with the new host name.
   b. Change ASI.ARMEAddresses to replace the old host name with the new host name.

c. Change `ASI.PDAddresses` to replace the old host name with the new host name.

8. Run `unenroll.bat secure` **before** you change the host name.
9. Modify *`BEA_HOME`*`/ales30-ssm/<`*`ssm`*`>/adm/ssm_install.properties` to replace the old host name with the new host name:

   a. If the host name of the Admin server is changed, change `admin.host`.

10. Modify *`BEA_HOME`*`/ales30-ssm/<`*`ssm`*`>/template/adm/ssm_instance.properties` to replace the old host name with the new host name:

    a. If the host name of the ALES Administration host is also changed, then change `admin.host = <`*`OLD-HOSTNAME`*`>` to `admin.host = <`*`NEW-HOSTNAME`*`>`.

11. Modify *`BEA_HOME`*`/ales30-ssm/<`*`ssm`*`>/template/config/WLESarme.properties` to replace the old host name with the new host name:

    a. Change `PDAddress = https://<`*`OLD-HOSTNAME`*`>` to `PDADDRESS = https://<`*`NEW-HOSTNAME`*`>`.

    b. Change `instanceName = ARME.....asi.<OLD-HOSTNAME>` to `instanceName = ARME....asi.<`*`NEW-HOSTNAME`*`>`.

12. Modify the policy database (Oracle and Sybase) to specify the new IP address.

    For Sybase, assuming a username of `wles`, a password of `password`, and a sybase server name of `MYHOST`, perform the following steps:

    a. Start one command window and run `isql -Dwles -Ppassword -SMYHOST`.

    b. Run `select * from engine_addresses` to verify the old address are being used.

    c. Delete the records in the `engnie_addresses` table that refer to the old address.

    For Oracle, the steps are similar to those for Sybase:

    a. `sqlplus wles/password@ASI`

    b. `select * from engine_address;`

    c. `truncate table engine_addresses;`

13. Run `enroll.bat secure` to enroll the SSM instance with the new host name.

Admin Server and SSM on Different System, Hostname of SSM Changed, SSM Instance in Demo Mode

In this scenario, the Admin server and the SSM are installed different systems.

Follow these steps to reconfigure ALES:

1. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/adm/sil30ent_install_ssm.xml as follows:
   a. Change <data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<*old ssm host name*>:7010/" />
   b. Change <data-value name="SCM_BACKUP_ADMIN_URL" value="https://<*old ssm host name*>:7010/" />
2. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/adm/ssm_instance.properties to replace the old host name with the new host name:
   a. Change host.name = <*OLD-HOSTNAME*> to host.name = <*NEW-HOSTNAME*>.
   b. If the host name of the ALES Administration host is also changed, then change admin.host = <*OLD-HOSTNAME*> to admin.host = <*NEW-HOSTNAME*>.
3. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*/ales30-ssm/<*ssm*>/*template*/bin/set-wls-env.bat to replace the old host name with the new host name:
   a. Change -Dwles.config.signer=<*OLD-HOSTNAME*> to -Dwles.config.signer=<*NEW-HOSTNAME*>.
4. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/template/WLESWeblogic.conf or WLESws.wrapper.conf to replace the old host name with the new host name.
5. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/template/config/WLESarme.properties to replace the old host name with the new host name:
   a. Change PDAddress = https://<*OLD-HOSTNAME*> to PDADDRESS = https://<*NEW-HOSTNAME*>.
   b. Change instanceName = ARME.....asi.<OLD-HOSTNAME> to instanceName = ARME....asi.<*NEW-HOSTNAME*>.

Admin Server and SSM on Different System, Hostname of SSM Changed, SSM Instance in Secure Mode

In this scenario, the Admin server and the SSM are installed different systems.

Follow these steps to reconfigure ALES:

1. Run `unenroll.bat secure` **before** you change the host name.
2. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/adm/silent_install_ssm.xml` as follows:
   a. Change `<data-value name="SCM_PRIMARY_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`
   b. Change `<data-value name="SCM_BACKUP_ADMIN_URL" value="https://<`*old ssm host name*`>:7010/" />`
3. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/adm/ssm_instance.properties` to replace the old host name with the new host name:
   a. Change `host.name = <`*OLD-HOSTNAME*`>` to `host.name = <`*NEW-HOSTNAME*`>`.
   b. If the host name of the ALES Administration host is also changed, then change `admin.host = <`*OLD-HOSTNAME*`>` to `admin.host = <`*NEW-HOSTNAME*`>`.
4. If the WLS 8.1 or WLS SSM, modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/bin/set-wls-env.bat` to replace the old host name with the new host name:
   a. Change `-Dwles.config.signer=<`*OLD-HOSTNAME*`>` to `-Dwles.config.signer=<`*NEW-HOSTNAME*`>`.
5. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/config/WLESWeblogic.conf` or `WLESws.wrapper.conf` to replace the old host name with the new host name.
6. Modify *BEA_HOME*`/ales30-ssm/<`*ssm*`>/`*template*`/config/WLESarme.properties` to replace the old host name with the new host name:
   a. Change `PDAddress = https://<`*OLD-HOSTNAME*`>` to `PDADDRESS = https://<`*NEW-HOSTNAME*`>`.
   b. Change `instanceName = ARME.....asi.<OLD-HOSTNAME>` to `instanceName = ARME....asi.<`*NEW-HOSTNAME*`>`.
7. Run `enroll.bat secure` to enroll the SSM instance with the new host name.

Configuration for a New IP Address

This section describes how to reconfigure ALES components if the IP address is subsequently changed. The steps you follow depend on how the ALES components are installed:

Host IP of Admin Server and SCM Changed, Host IP of SSM Not Changed

If the host IP of the Admin server and SCM is changed but the host IP of the SSM is not changed, follow all of these steps.

1. Shut down all ALES services, including the Admin server and the SCM.

   a. *BEA_HOME*/ales30-admin/bin/WLESadmin.sh stop

   b. *BEA_HOME*/ales30-scm/bin/WLESscm.sh stop

   c. Stop SSM instance.

   Note: If any of these components were started in console mode, you may need to type CTRL-C in a Command window to stop them.

2. Modify *BEA_HOME*/ales30-scm/config/SCM.properties to replace the old IP address with the new IP address in the following line:

   ```
   OS.interface = 10.120.3.140
   ```

3. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/adm/ssm_install.properties to replace the old IP address with the new IP address in the following line:

   ```
   scm.interface.list = 10.120.3.140
   ```

4. Restart the SCM , the Admin server, and the SSM.

SSM IP is Changed, IP of Admin Server and SCM Not Changed

If the host IP of the SSM is changed, follow these steps.

1. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/adm/ssm_install.properties to replace the old IP address with the new IP address in the following line:

   ```
   scm.interface.list = 10.120.3.140
   ```

2. Restart the SCM , the Admin server, and the SSM.

SSM is on Same Host as Admin Server and SCM

If the Admin server, SCM, and SSM are all on the same host and the IP address is changed, follow these steps.

1. Shut down all ALES services, including the Admin server and the SCM.

   a. *BEA_HOME*/ales30-admin/bin/WLESadmin.sh stop

   b. *BEA_HOME*/ales30-scm/bin/WLESscm.sh stop

   c. Stop SSM instance.

   Note: If any of these components were started in console mode, you may need to type CTRL-C in a Command window to stop them.

2. Modify *BEA_HOME*/ales30-scm/config/SCM.properties to replace the old IP address with the new IP address in the following line:

   ```
   OS.interface = 10.120.3.140
   ```

3. Modify *BEA_HOME*/ales30-ssm/<*ssm*>/adm/ssm_install.properties to replace the old IP address with the new IP address in the following line:

   ```
   scm.interface.list = 10.120.3.140
   ```

4. Restart the SCM , the Admin server, and the SSM.

CHAPTER 6

Configuring SSL in the Web Services SSM

When you create a Web Services SSM instance, the SSM is accessible via HTTP. This is appropriate for development and for debugging purposes, but production environments should use one-way SSL or two-way SSL (SSL with client authentication).

This section describes how to enable one-way SSL communication between a Web Services SSM and its client. It is assumed that the reader has basic knowledge of the SSL protocol, Certificate Authorities (CA), X.509 certificates and Java Key Stores (JKS).

In this section, `%SSM_INST_HOME%` represents the installation folder of the Web Services SSM, for example, `c:\bea\ales30-ssm\webservice-ssm\instance\wsssm`.

Configuring One-Way SSL

With one-way SSL, the SSM sends its identity certificate to the client, therefore the client must trust the CA that signed the identity certificate. (The client does not have to have its own certificate, because it is not authenticated by the Web Services SSM.)

To configure a Web Services SSM to use one-way SSL:

1. Stop the Web Services SSM if it is running.
2. Delete the contents of the `%SSM_INST_HOME%\apps` directory.
3. Run the following command to regenerate the content of the `apps` directory:

   ```
   %SSM_INST_HOME%\adm\ssmwsInstance.bat -m
   ```

4. Restart the Web Services SSM.

The client's trusted JKS is defined by the system property `javax.net.ssl.trustStore` and the JKS password is defined by the system property `javax.net.ssl.trustStorePassword` property. (These properties are defined in the Java Secure Socket Extension (JSSE) documentation.) You can specify these system properties by running the Web Services SSM Java client with command line arguments such as:

```
-Djavax.net.ssl.trustStore=C:\jks\trust.jks
-Djavax.net.ssl.trustStorePassword="secretword"
```

For testing purposes, the client can use the `%SSM_INST_HOME%\ssl\trust.jks` JKS, which contains the CA that was used to sign the default server's identity.

Adding New Identity Assertion Types

To add support for new assertion types to the Web Services SSM:

1. Create a new Java class as a holder for the identity assertion. Note that the new holder class must belong to the `com.bea.security.ssmws.credentials` namespace. In this procedure, we use a class named `com.bea.security.ssmws.credentials.TestCredHolderImpl` and a custom identity assertion type named `TestIA`. See Figure 6-1 for an example.

2. Add the JAR file containing the holder class to the Web Service SSM's classpath. To do this, modify the `WLESws.wrapper.conf` configuration file located in `BEA_HOME/ales30-ssm/webservice-ssm/instance-name/config`. For example, if the holder class is contained in a file named `ssmwsCustomAssertion.jar`, add a line like this to `WLESws.wrapper.conf`:

   ```
   wrapper.java.classpath.40=C:/bea/ales30-ssm/webservice-ssm/lib/ssmwsCus
   tomAssertion.jar
   ```

   Note: The `wrapper.java.classpath` lines must increment sequentially.

3. Modify the mapping file for incoming messages. Mapping for incoming messages is controlled by the `castor.xml` file in the `BEA_HOME/ales30-ssm/webservice-ssm/lib/com/bea/security/ssmws/soap` directory. Add an entry like the following inside the <mapping> XML element:

   ```
   <class name="com.bea.security.ssmws.credentials.TestCredHolderImpl">
       <map-to cst:xml="TestIA" />
       <field name="cookie" type="java.lang.String" >
           <bind-xml node="text"/>
       </field>
   </class>
   ```

4. Modify the mapping file for outgoing messages. Mapping for incoming messages is controlled by the `castor.xml` file in the `BEA_HOME/ales30-ssm/webservice-ssm/lib/com/bea/security/ssmws/credentials` directory. Add an entry like the following inside the <mapping> XML element

```
<class name="com.bea.security.ssmws.credentials.TestCredHolderImpl">
<map-to cst:xml="TestIA"
cst:ns-uri="http://security.bea.com/ssmws/ssm-soap-types-1.0.xsd" />
<field name="cookie" type="java.lang.String" >
    <bind-xml node="text"/>
</field>
</class>
```

5. To log SOAP messages received and sent by the Web Services SSM, make the following changes to the SSM instance's `config/log4j.properties` file:

   a. Change `log4j.appender.A1.Threshold=ERROR` to `log4j.appender.A1.Threshold=DEBUG`

   b. Add the following entry: `log4j.logger.com.bea.security.ssmws.server=DEBUG`

When the Web Services SSM is started, it will use the new holder implementation and the mapping entries to convert back and forth between the token's XML and Java representations.

Figure 6-1 Sample Identity Assertion Holder Class

```
public class TestCredHolderImpl implements CredentialHolder
{
private String m_cookie;
public static final String m_Type = "TestIA";

public void setCookie(String cookie)
{
    m_cookie = cookie;
}
public String getCookie()
{
    return m_cookie;
}
public Object getObject()
{
    return getCookie();
}
public void setObject(Object cred)
{
    setCookie((String)cred);
}
public String getType()
```

```
{
    return TestCredentialHolderImpl.m_Type;
}
public String getAsString()
{
    return m_cookie;
}
}
```

CHAPTER 7

Database Password Changes

This section describes how to change the account password used to access the authentication database or the ASI Authorizer metadirectory database.

Note: See *Administration Server Installation Guide* for instructions on changing the Administrator default password.

Auth Provider Password (WLS SSM)

1. Modify the password for the ALES user in the database.
2. Log in to the WebLogic Server console for the `asiDomain`.
3. Specify the new password in the **Database User Password** field under the **Security Realms > asiadmin > Providers > ALES Database Authenticator > Provider Specific** tab.

   Set the Database User Password to match the password you used in Step 1. This must be associated with a valid database login for your database with read access to the tables that involve the authentication process.
4. Click on **Activate Changes**.
5. Run the asipassword utility.
6. Restart the ALES Administration Server.

Auth Provider Password (WLS 8.1 SSM)

1. Modify the password for the ALES database user in the database.
2. Change the password in the ALES Administration Console:
   a. Open the SSM configuration where the provider is defined.
   b. On the **Authentication Providers** tab, select the **Database Authentication** provider.
   c. Click the **Details** tab and set the **Database User Password** to match the password used in Step 1. This must be associated with a valid database login for your database with *Read* access to the tables that involve the authentication process.

      Enter the same password in the confirmation field.
   d. Click **Apply** to save your changes.

      Note: Note: Changes made to a provider do not take effect until after it is explicitly deployed and the associated SSM is restarted.
3. Run the asipassword utility.
4. Restart the Administration Server.

ASI Authorizer Password (WLS SSM)

1. Modify the password for the ALES user in the database.
2. Log in to the WebLogic Server console for the **asiDomain**.
3. Specify the new password in the Database User Password field under **Security Realms** > **asiadmin** > **Providers** > **ASI Authorizer** > **Provider-Specific** tab.
4. Click **Activate Changes**.

ASI Authorizer Password (WLS 8.1 SSM)

1. Modify the password for the metadirectory database user in the database.
2. Open the SSM configuration where the provider is defined.
3. On the **Authorization** folder, click **Authorization** and select the **ASI Authorization** provider.

4. Click the **Details** tab and change the **Database Login Password** field for the metadirectory database user.
5. Confirm the password change and click **Apply**.

   Note: Changes made to a provider do not take effect until after it is explicitly deployed and the associated SSM is restarted.
6. Run the asipassword utility.
7. Restart the Administration Server.

CHAPTER 8

Resetting the ALES Administrator Password

In the event the ALES administrator password is lost, the `generatePasswordHash` (`bat` or `sh`) utility can be used to reset it and restore access to the Administration Console and Entitlements Management Tool.

Procedure

Follow these steps to reset the ALES administrator password:

1. To generate a hashed version of the new administrator password, open a command window in *`BEA_HOME`*`/ales30-admin/bin` and enter the following:

   `generatePasswordHash.bat <`*`new_password`*`>`

   This generates a hashed password. In the following example, the hashed password is shown inside square brackets.

   `hash result is[{SHA1}pvGBjCW7IS5jCM1e9dYR/EtCTojHjqk=]`

2. To update the database table for administration user, do the following:

   a. Connect to the database and the schema defined during ALES installation.
   Note: This can be obtained by examining the following file:

   `BEA_HOME\ales30-admin\config\database.properties`.

   b. Enter the following:

   ```
   SQL> update adminuser set password = '<hash_password>' where userid =
   '//user/asi/system/';
   ```

where
<*hash_password*> is the hashed password generated in step 1.

Example:

```
SQL> update adminuser set password =
'{SHA1}pvGBjCW7IS5jCM1e9dYR/EtCTojHjqk=' where userid =
'//user/asi/system/';
```

3. To establish the new password and update the `password.xml/password.key` file, open a command window in *BEA_HOME*`/ales30-admin/bin` and execute:

   ```
   asipassword.bat <admin_username>
   <BEA_Home>\ales30-shared>\keys\password.xml
   <BEA_Home>\ales30-shared>\keys\password.key
   ```

   Note: This entry is one line.

   where

   `<admin_username>` is the ALES administrator username (be default, `system`)
   <BEA_Home> is BEA_HOME, for example `c:\bea`

4. Start the Administration Server.

CHAPTER 9

Resource Discovery

The most challenging aspect of writing policy for an application is discovering all the application resources that must be secured. This process is greatly simplified by running the SSM in 'discovery' mode and then performing one or more user sessions that reflect actual use in the application. Based on the activities performed during the user session, ALES will generate an initial policy set to files that can then be imported into ALES.

Note: Do not use discovery mode in a production environment. Use it only during development to create the initial security policy.

Enabling Discovery Mode

Resource discovery is enabled by setting the ASI Authorization and ASI Role Mapping providers to run in discovery mode. In this mode, these providers always return 'true' when evaluating user requests and generate the initial policy files based on those requests.

To enable discovery mode, modify the command line that starts the SSM by adding the following system properties:

```
com.bea.security.providers.authorization.asi.AuthorizationProviderImpl.discoverymode=true
```

```
com.bea.security.providers.authorization.asi.RoleProviderImpl.discoverymode=true
```

The system properties are set using the -D switch in the appropriate file. As an example, to enable resource discovery for the WLS SSM, add the following lines to the SSM's `set-wls-env.bat` file:

```
set WLES_JAVA_OPTIONS=%WLES_JAVA_OPTIONS%
-Dcom.bea.security.providers.authorization.asi.AuthorizationProviderImpl.d
iscoverymode=true
set WLES_JAVA_OPTIONS=%WLES_JAVA_OPTIONS%
-Dcom.bea.security.providers.authorization.asi.RoleProviderImpl.discoverym
ode=true
```

For each SSM, Table 9-1 indicates the name and location of the file that must be modified.

Table 9-1 Setting System Properties for Discovery Mode

SSM Type	File Name	Default Location
Java	`set-env.bat (.sh)`	*BEA_HOME*`\ales30-ssm\java-ssm\instance\<`*instancename*`>\bin`
Web Services	`wlesws.wrapper.conf`	*BEA_HOME*`\ales30-ssm\webservice-ssm\instance\<`*instancename*`>\config`
WebLogic Server 8.1	`set-wls-env.bat (.sh)`	*BEA_HOME*`\ales30-ssm\wls8-ssm\instance\<`*instancename*`>\bin`
WebLogic Server 9.x/10.x	`set-wls-env.bat (.sh)`	*BEA_HOME*`\ales30-ssm\wls-ssm\instance\<`*instancename*`>\bin`

Running in Discovery Mode

After enabling discovery mode as described in the previous section, start the secured application. Then perform a user session by logging in to the application, exercising requests for resources, and invoking application functions.

It is important to note that the generated files are meant to serve as a starting point for defining a policy set to fully secure the application. In particular, note the following:

- The recorded policy data is based only on requests made during the user session; no policy data will be generated for parts of the application that are not used.
- Depending on the Resource hierarchy you use to define the application's resources, the imported policy may contain more Resources than actually needed. This can be a particular problem when securing a WLP application. Unlike other applications, access to a WLP resource is denied if not explicitly granted.

When generating the files, user requests are transformed into a policy import format. Under this format, a request consists of four elements: Subject, Resource, Action, Attributes. Each element

has different restrictions on the allowable character set. The providers automatically normalize any invalid characters to produce a valid entry. See Character Restrictions in Policy Data for further details.

Importing the Policy

The files generated by discovery mode will be located in the SSM's domain directory. To import them into ALES, use the Policy Import tool.

Once imported, the policy can be managed using the Entitlements Management Tool.

CHAPTER 10

Running the Java SSM in Java Development Environments

The section describes how to run the Java SSM in WebSphere RAD and Eclipse development environments. This includes instructions for setting up sample applications that are provided when the Java SSM is installed.

WebSphere RAD Environments

This section demonstrates how to run the Java SSM in WebSphere RAD using the sample application provided in `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample`.

These steps were performed using WebSphere RAD 6.0.0 with the pre-installed IBM JDK.

1. Follow the `README` in `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample` to setup the sample policies for this example.
2. Start WebSphere RAD in a new workspace and create a new Java project.
3. Right-click the project and select **Import** to import the Java file from the `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample\src` directory into the new project.
4. Add all of the ALES libraries used by the `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample\config\run.bat` file to the **Java Build Path** in the Project Properties. To do this:

a. Right-click the project and select **Properties**.

b. Click **Java Build Path**.

c. On the **Libraries** tab, click **Add External JARs** and add the jar files located in the following directories:

`BEA_HOME\ales30-ssm\java-ssm\lib` (except for `pdsoap1.jar`)
`BEA_HOME\ales30-ssm\java-ssm\lib\providers\ales`

d. Click **OK** and allow the project to build. All errors should be resolved.

5. To setup the arguments for the `main()` method, right-click the Java file and select **Run**. Then select **Run** with the green arrow icon.

6. In the **Run** dialog, select **Java Application**. Then click **New** and supply a well-defined name for the configuration.

7. Click **Search** to find the Java class with the main method. In this example it would be `com.bea.security.examples.JavaAPIExample`.

8. On the **Arguments** tab, paste in the command line options used by run.bat (in the `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample`) into the **VM arguments** field.

   Example of the settings used:

```
-Dwles.scm.port=7013 -Dwles.arme.port=8100
-Dwles.config.signer=<HOSTNAME>
-Dlog4j.configuration="file:./java-ssm/instance/jssm/config/log4j.prope
rties" -Dlog4j.ignoreTCL=true
-Dwles.ssl.passwordFile="C:/bea/ales30-shared/keys/password.xml"
-Dwles.ssl.passwordKeyFile="C:/bea/ales30-shared/keys/password.key"
-Dwles.ssl.identityKeyStore="C:/bea/ales30-shared/keys/identity.jceks"
-Dwles.ssl.identityKeyAlias=wles-ssm
-Dwles.ssl.identityKeyPasswordAlias=wles-ssm
-Dwles.ssl.trustedCAKeyStore="C:/bea/ales30-shared/keys/trust.jks"
-Dwles.ssl.trustedPeerKeyStore="C:/bea/ales30-shared/keys/peer.jks"
-Djava.io.tmpdir="./java-ssm/instance/jssm/work/jar_temp"
-Darme.configuration="./java-ssm/instance/jssm/config/WLESarme.properti
es" -Dales.blm.home="./java-ssm/instance/jssm" -Dkodo.Log=log4j
-Dwles.scm.useSSL=true -Dwles.providers.dir=./java-ssm/lib/providers
```

9. While in the **Arguments** tab, clear the **Use default working directory** checkbox.

10. Select the **File System** and browse to the build directory of the JavaAPIExample.

For example,
`BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample\build\config`.

11. On the **Classpath** tab, select the **User Entries** node on the Classpaths tree. Then click the **Advanced** button and select **Add External Folder** and add the following external folders:

    ```
    BEA_HOME
    BEA_HOME\ales30-ssm\java-ssm\instance\jssm\config
    ```

12. Click **Apply** and then **Run**.

    NOTE: If an exception like the following appears...

    ```
    com.bea.security.management.ConfigurationException: Error initializing
    the SCM SSL context. at
    com.bea.security.internal.css.SCMConfiguration.configureRealm(SCMConfig
    uration.java:512)
    ```

    ...then do the following to switch to a standard JDK installation:

    a. Select **Window->Preferences->Java -> Installed JREs**.

    b. Select **add** and add a JVM (Sun or BEA JRockit).

    c. Select **OK** and check the newly added JVM.

13. Enter in the arguments to see the example correctly authorizing a subject on a resource.

Troubleshooting

To troubleshoot any problems, enable verbose debugging on the Java-SSM and examine the log for the authorization events. To do this:

1. Open `logj.properties` in the SSM instance's `config` directory in an editor and uncomment the following lines:

   ```
   log4j.logger.com.bea.security.providers.authorization = DEBUG
   log4j.logger.com.wles.util.DebugStore=DEBUG
   ```

2. Clear out the log files in the SSM instance's `log` directory.

3. Re-run the Java SSM example and examine `system_console.log` in the instance's `log` directory for authorization events, such as:

   ```
   2008-04-14 15:03:28,343 [main] DEBUG
   com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
   or - evaluateRuleArray(): Evaluate policy:
   4405:grant(//priv/buy,//app/policy/jssm/store/book,//role/borrower)
   2008-04-14 15:03:28,343 [main] DEBUG
   ```

```
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - boolEvaluate() entered for rule with condition: true
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - constraint evaluation result is: true
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - append return attributes.
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - boolEvaluate(): Evaluation result size is 0.
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - evaluateRuleArray(): Evaluate result: true
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - evaluateGrantPolicy result: true
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.evaluator.BoolEvaluat
or - authEvalWorker: evalute with roles return GRANT
2008-04-14 15:03:28,343 [main] DEBUG com.wles.util.DebugStore -
queryAccess: DebugStore:

========== Policy Evaluation Info ==========
RequestResource is: //app/policy/jssm/store/book
UserInfo:
    Name: //user/asi/system/
    Groups: //sgrp/asi/allusers/
Resource Present: true
Roles Granted: //role/borrower //role/EntitlementsAdmin
Role Mapping Policies:
    1. Result: true; Policy Type: grant
        Role: //role/borrower
        Resource: //app/policy/jssm/store
        Subject: //user/asi/system/
        Constraints: canbuy="yes"
        Evaluated Attributes and Functions:
            sys_user(dynamic) = system
            canbuy(dynamic) = yes

    2. Result: true; Policy Type: grant
        Role: //role/EntitlementsAdmin
        Resource: //app/policy
        Subject: //user/asi/system/
        Constraints: NONE

ATZ Policies:
    1. Result: true; Policy Type: grant
        Privilege: //priv/buy
```

```
        Resource: //app/policy/jssm/store/book
        Subject: //role/borrower
        Constraints: NONE

========== Policy Evaluation Info ==========
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.ARME.engine.ARME - unlock
policy lock for read
2008-04-14 15:03:28,343 [main] DEBUG
com.bea.security.providers.authorization.asi.AuthorizationProviderImpl
- result is GRANT
2008-04-14 15:03:28,359 [main] DEBUG
com.bea.security.providers.authorization.asi.AccessResultLogger -
Subject Subject:
Principal: system
 privilege buy resource //app/policy/jssm/store/book result PERMIT
```

Eclipse

This section demonstrates how to run the Java SSM in Eclipse using the sample application provided in `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample`.

1. In Eclipse, create a Java project from the sample source files found in the following directory:

   `BEA_HOME\ales30-ssm\java-ssm\examples\JavaAPIExample`

2. Set the classpath in Java project based on `set-env.bat` of SSM instance. Place the following external jar files into "Java Build Path" of Eclipse project

   ```
   saaj.jar
   api.jar
   css.jar
   log4j.jar
   scmapi.jar
   framework.jar
   jsafeJCEFIPS.jar
   asi_classes.jar
   connector.jar
   EccpressoCore.jar
   EccpressoJcae.jar
   jmx.jar
   jsafeFIPS.jar
   asitools.jar
   ssladapter.jar
   sslplus.jar
   wlcipher.jar
   antlr.jar
   ```

```
javax.servlet.jar
org.apache.jasper.jar
org.mortbay.jetty.jar
pdsoap.jar
process.jar
sslclient.jar
sslserver.jar
webservice.jar
webserviceclient.jar
ld-server-core.jar
ld-client.jar
wlsdo.jar
wlxbean.jar
xbean.jar
xqrl.jar
jaxrpc.jar
wsdl4j-1.5.1.jar
axis.jar
commons-discovery-0.2.jar
commons-logging-1.0.4.jar
providers/ales/serp.jar
providers/ales/commons-collections-3.2.jar
providers/ales/commons-lang-2.1.jar
providers/ales/commons-pool-1.3.jar
providers/ales/CR338979_414_jdk1.4.jar
providers/ales/jdo.jar
providers/ales/jta-spec1_0_1.jar
providers/ales/kodo-runtime.jar
providers/ales/openjpa.jar
```

3. Add the `BEA_HOME` and `BEA_HOME\ales30-ssm\java-ssm\<`*`instance`*`>\config` directories into the Eclipse classpath.

   Note: If Eclipse reports, "*Cannot not nest the directory inside library <BEA-HOME>* ", copy `license.bea` into the instances `config` directory.

4. Set the following run configurations in the project:

   – Working directory — `BEA_HOME\ales30-ssm\java-ssm\example\JavaAPIExample\build\config`

   – VM arguments based on %JAVA-OPTIONS% of `set-env.bat`.

Troubleshooting

Note the following error conditions and resolution:

AXIS SOAP compatibility issue:
'java.lang.NoSuchFieldError: RPC'
The issue is caused by AXIS SOAP stack compatibility between different AXIS version. Delete `pdsoap1.jar` from classpath.

License Check:
'Got exception in reading the license file'.
Make sure `license.bea` file is in classpath.

XML Parsing:
'java.lang.NoSuchMethodException: org.apache.axis.encoding.ser.ArraySerializerFactory.create(java.lang.Class, javax.xml.namespace.QName)'
Only the jar files set in `set-env.bat` should be included in the project. Files like the following are not needed and should be removed:

- `com.bea.core.common.security.opensaml2_4.0.0.0.jar`,
- `com.bea.core.xml.beaxmlbeans_2.2.0.0.jar`
- `javax.xml.stream_1.0.0.0.jar`
- `xml-apis.jar`

CHAPTER 11

Debugging Policies

This document desscribes how to set an SSM instance's logs to record debugging-level events having to do with authentication, role mapping, and authorization.

Overview

When policy outcomes are other than expected, it may be useful to enable policy debugging so that the SSM's logs will capture all events related to policy decisions. The logged information may policy-related details, such as failed authentications, missing group memberships, incorrect role assignments, and others.

Caution: SSM performance can be severely impacted when debug flags are enable. In production environments turn on debug flags only when necessary.

Enabling Policy Debugging

Policy debugging is enabled by changing settings in the SSM instance's `log4j.properties` files. To do this:

1. In the SSM instance's `config` directory, open `log4j.properties` in an editor.
2. To turn on authentication debugging, uncomment the following line:

   ```
   log4j.logger.com.bea.security.providers.authentication = DEBUG
   ```

3. To turn on role mapping and authorization debugging, uncomment the following lines:

   ```
   log4j.logger.com.bea.security.providers.rolemapper = DEBUG
   log4j.logger.com.bea.security.providers.authorization = DEBUG
   log4j.logger.com.wles.util.DebugStore=DEBUG
   ```

4. Restart the SSM.

Event Logs

This section describes common policy-related events that may be captured when in debugging mode.

Authentication

For authentication events, check for the events shown in Table 11-1:

Table 11-1 Authentication Events

Event	Description
Usernames	Usernames should match those supplied to the SSM. Usernames are logged as follows: `DBMSAtnLoginModuleImpl - Login username: <username>`
Identity Directory	Check that the identity directory name is correct. Directory names are logged as follows: `DefaultDBMSPluginImpl - Formatted User: //user/<directory>/<username>/`

Table 11-1 Authentication Events

Event	Description
Authentications	The following message indicates a successful authentication: `DBMSAtnLoginModuleImpl - Authenticated User <username>`
Groups	The following message indicates a user's group memberships: `odbms.DBMSAtnLoginModuleImpl - Groups Found: <list-of-groups>`

Role Mapping

For role mapping events, check for the events shown in Table 11-2:

Table 11-2 Role Mapping Events

Event	Description
Roles	The following entry denotes the entry point for evaluating the roles. <*username*> is the user name supplied to the application and <*resource*> is the name of the queried resource. `BoolEvaluator - Query roles entered for //user/asi/<username>/://app/policy/<resource>`
Role Policies	Make sure all relevant policies are evaluated. The following is a sample logged event: `BoolEvaluator - evaluateGrantDenyRoles: evalute grant policy: 3600:grant (//role/<role>,//app/policy/<resource>, //user/<username>/)`
Constraint Evaluations	The following is a sample message indicating a constraint evaluation: `BoolEvaluator - constraint evaluation result is: true`
Roles Granted	The following is a sample message indicating a role assignment: `BoolEvaluator - Role //role/<role> was granted`

Authorization

For authorization events, check for the events shown in Table 11-3:

Table 11-3 Authorization Events

Event	Description

Authorization	The following entry indicates the authorization policy evaluated:
	`BoolEvaluator - evaluateRuleArray():`
	`Evaluate policy: 3401:grant`
	`(//priv/buy,//app/policy/javaapi_app/store/book,`
	`//role/borrower)`

Constraint Evaluations	The following is a sample message indicating a constraint evaluation: `BoolEvaluator-constraint evaluation result: true`

Roles Granted	The following is a sample message indicating a authorization policy evaluation:
	`BoolEvaluator - authEvalWorker:`
	`evalute with roles return GRANT`

Sample Log Messages

========== Policy Evaluation Info ==========
RequestResource is: //app/policy/<resource>
UserInfo:
Name: //user/<identity-directory>/<user-name>
Groups: //sgrp/<identity-directory>/<group-name>

Resource Present: true
Roles Granted: //role/<granted-roles>

Role Mapping Policies:
1. Result: true; Policy Type: grant
Role: //role/<requested-role>
Resource: //app/policy/<resource>
Subject: //user/<identity-directory>/<user-name>
Constraints: (some-variable = "some-value")
Evaluated Attributes and Functions:
some-variable(identity) = some-value

ATZ Policies:
1. Result: true; Policy Type: grant
   Privilege: //priv/<requested-privilege>
   Resource: //app/policy/<resource>
   Subject: //role/<granted-role>
   Constraints: NONE

========== Policy Evaluation Info ==========

Debug API for Java-SSM

The following 2 API calls are specific to Java-SSM only. To enable policy debugging, open `BEA_HOME/ales30-ssm/java-ssm/<instancename>/jssm/config/WLESarme.properties` in an editor and set the following:

```
SsmPolicyTrace=true
```

Note: Enabling debugging weakens security, because OES policy evaluations will be visible to Java programs. For example, a malicious Java program could make "_Debug()" calls to gain information about policies.

To capture debugging data:

1. Create a DebugInfo object as follows:

   ```
   DebugInfo debugInfo = new DebugInfo();
   ```

2. Call getRoles_Debug() to obtain role assignments, as follows:

   ```
   getRoles_Debug(AuthenticIdentity ident, RuntimeResource resource,
   RuntimeAction action, AppContext context, DebugInfo debugInfo)
   ```

3. Call isAccessAllowed_Debug() to obtain information about the policies used to reach a decision, as follows:

   ```
   isAccessAllowed_Debug(AuthenticIdentity ident, RuntimeResource
   resource, RuntimeAction action, AppContext context, DebugInfo
   debugInfo)
   ```

4. Print the DebugInfo object to console:

   ```
   System.out.println(debugInfo.toString());
   ```

For further information, see the sample provided in the `BEA_Home/ales32-ssm/java-ssm/examples/JavaAPIExample` directory.