Chapter 7

Configuring LDAPS for SRC Components

LDAPS is LDAP that uses secure sockets layer (SSL) to secure communications between an LDAP client and server. You can configure particular SAE components to use LDAPS to connect to the directory.

Topics in this chapter include:

Overview of LDAPS Support

The SAE supports LDAPS connections to the directory server for its components, and can provide simultaneous LDAP and LDAPS connections for different components. You can configure the SAE to use LDAPS for some directory connections and LDAP for other directory connections. When planning whether to use an LDAP or LDAPS connection, consider that LDAPS connections have higher processing requirements, use more network bandwidth, and are slower than LDAP connections.

LDAPS connections are useful for protecting confidential data such as attributes that contain passwords and keys. For example, if you want data exchanged between a component such as User Data Manager and the directory to be more secure, you can configure the connection to use LDAPS. For public data that does not require the security of SSL (such as a directory connection that transmits only service information), you can configure LDAP rather than LDAPS.

Most directories, including Oracle Internet Directory, Sun ONE Directory Server, and DirX support LDAP connections through SSL.

LDAPS Authentication and Connection

The steps in the LDAPS authentication and connection sequence are:

1. The directory client initiates LDAPS connection.
2. The directory server sends the X.509 SSL server certificate that it has received from a certificate authority (CA).

3. The client checks the certificate against its trust certificate store. If it matches, the certificate is trusted.

4. The client proceeds with establishing the SSL connection.

5. When the SSL connection is up, the client sends a bind DN and password to the server to establish the LDAP connection.

6. The server authenticates the client and establishes the LDAP over SSL connection.

NOTE: The SRC software does not support certificate authentication for directory clients.

Configuring LDAPS Connections

The tasks to configure LDAPS connections are:

1. Configuring the Directory Server to Support LDAPS Connections on page 62

2. Establishing Trust for Directory Clients on page 63

3. Configuring the SAE to Find the Certificate Store on page 63

4. Enabling LDAPS Communication for SAE Components on page 64

5. Disabling LDAPS Communication for SAE Components on page 65

Configuring the Directory Server to Support LDAPS Connections

For information about how to perform these tasks, see the documentation for your directory server.

For the SAE to communicate with a directory over LDAPS, typically you must configure your directory server to support SSL connections by:

- Obtaining a signed certificate for the directory server from a CA.

  There are many well-known CAs. You can also set up your own CA to sign the directory certificate. The CA must be trusted by the directory clients that use LDAPS to communicate with the directory. Tools such as OpenSSL (http://www.openldap.org) are available to set up a CA.

- Setting up the directory server with an X.509 SSL server certificate. Typically, you install a certificate for the server, and configure the directory server to trust the CA's certificate.

- Enabling SSL.

Establishing Trust for Directory Clients

Each directory client must have a certificate database and must trust the CA to use SSL connections to the directory server.

The SAE, like other Java applications, implicitly trusts certificates that are stored in the */jre/lib/security/cacerts* certificate file. This file is a Java Runtime Environment (JRE) systemwide certificate trust store. By default, the file contains certificates from well-known CAs.

If a certificate for the CA that you use for the directory server is available in the *cacerts* file:

- View the file on the host on which you installed the JRE.

If your CA is not in the *cacerts* file:

- Import the CA into this file or into any certificate store that is in Java Keystores (JKS) format (supported by the Java 2 Software Development Kit). All Java applications running in a specified JRE trust all CAs present in the *cacerts* file.

You can also store a CA certificate in a location other than the default *cacerts* file. You might consider storing the CA elsewhere if you want your SAE to trust only the certificate for the CA that signs the directory server's certificate, or if you do not want other applications that are running in the same JRE to trust the CA's certificate.

To import a CA certificate into a store other than the default *cacerts* file:

- Use the Java **keytool** command.

The following example imports the CA's certificate *ca.crt* into a trust store named *ldapclient.keystore*.

```
keytool -import -v -trustcacerts -alias saeldap -noprompt -file ./ca.crt -keystore
ldapclient.keystore -storepass zaqwsx
```

For more information about the **keytool** command, see

http://java.sun.com/j2se/1.4.1/docs/tooldocs/solaris/keytool.html

Configuring the SAE to Find the Certificate Store

To enable the SAE to locate the certificate store, edit the */opt/UMC/sae/etc/default.properties* file.

To use a certificate file other than the default:

- In the */opt/UMC/sae/etc/default.properties* file, specify the name and path of the file in the Security.ssl.trustcertstore property.

  The following example specifies that the SAE use the *trustcerts* file:

  ```
  Security.ssl.trustcertstore = /opt/UMC/sae/etc/trustcerts
  ```

To specify that the SAE use the default *cacerts* file:

- In the */opt/UMC/sae/etc/default.properties* file, add a comment character before the Security.ssl.trustcertstore property.

Enabling LDAPS Communication for SAE Components

To enable an LDAPS connection for an SAE component, you edit the security properties for the component. How you enable the properties depends on the component for which you are enabling LDAPS.

To enable an LDAPS connection for a component:

1. Open the configuration for the security properties for the component.

   Table 10 shows how to access the security properties for the various components.

Table 10: How to Access Security Properties for SRC Components

SRC Component	How to Enable Security Properties
■ Configuration Manager	■ Edit the */opt/UMC/sae/etc/default.properties* file.
■ User Data Manager ■ Equipment Data Manager ■ Service Data Manager ■ LDAP Authentication Plug-in ■ License Manager	1. In SDX Admin, select a configuration object (such as *l = POP_ID*) under *I = SAE, ou = staticConfiguration, ou = Configuration, o = Management, o = umc*. 2. Select the Main tab. The security properties appear in the list of properties on the Main tab.
■ Enterprise Service Portal User Data Manager ■ Enterprise Service Portal Service Data Manager	1. In SDX Admin, select *I = EASP, ou = staticConfiguration, ou = Configuration, o = Management, o = umc.* 2. Select the Main tab. The security properties appear in the list of properties on the Main tab.

2. Remove the comment character (#) that appears before the component's security protocol property. See Table 11.

Table 11: Security Protocol Properties for SAE Components

SAE Component	Security Protocol Property
Configuration Manager	Conf.directory.security.protocol
User Data Manager	UserDataSource.repository.ldap.server.security.protocol
Equipment Data Manager	UserCacheDataSource.repository.ldap.server.security.protocol
Service Data Manager	ServiceDataSource.repository.ldap.server.security.protocol
LDAP Authentication Plug-In	Plugin.ldapAuth.securityProtocol
License Manager	LicenseMgr.repository.ldap.server.security.protocol
Enterprise Service Portal User Data Manager	ent.repository.ldap.subscriber.manager.security.protocol
Enterprise Service Portal Service Data Manager	ent.repository.ldap.service.manager.security.protocol

If there is no comment character at the beginning of the line, the property is already enabled.

3. Set the server port property (as listed in Table 12) to the value supported for the LDAPS connection.

Table 12: Server Port Properties for SAE Components

SAE Component	Server Port Property
Configuration Manager	Conf.directory.port
User Data Manager	UserDataSource.repository.ldap.server.port
Equipment Data Manager	UserCacheDataSource.repository.ldap.server.port
Service Data Manager	ServiceDataSource.repository.ldap.server.port
License Manager	LicenseMgr.repository.ldap.server.port
Enterprise Service Portal User Data Manager	ent.repository.ldap.subscriber.server.port
Enterprise Service Portal Service Data Manager	ent.repository.ldap.service.server.port

For LDAPS connections, the default port number is 636.

4. Save the configuration.

Disabling LDAPS Communication for SAE Components

To disable an LDAPS connection for that component:

1. Open the configuration for the security properties for the component. See Table 10.

2. Add a comment character before the component's security protocol property. See Table 11.

3. Set the server port property (as listed in Table 12) to the value supported for the LDAP connection.

   For LDAP connections, the default port number is 389.

4. Save the configuration.