Using LDAP with Oracle® Java CAPS

Part No: 821–2546
March 2011

ORACLE

120126@25097

Contents

Using LDAP with Java CAPS

The topics listed here provide information about how to use the Lightweight Directory Access Protocol (LDAP) with Oracle Java Composite Application Platform Suite (Java CAPS).

For a list of LDAP servers supported by Java CAPS, see "Java CAPS 6.3 Components and Supported External Systems" in *Planning for Oracle Java CAPS 6.3 Installation* .

LDAP Overview

The Lightweight Directory Access Protocol (LDAP) is a standard that enables clients to query and update data in directory services.

An LDAP directory includes a series of *entries*. An entry is a collection of *attributes*, plus a *Distinguished Name* that uniquely identifies the entry.

In the following example, the first line specifies the DN. The succeeding lines specify the attributes.

```
dn: cn=all, ou=Roles, dc=company, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: Roles
```

The components of a DN are ordered hierarchically from most specific to least specific. Thus, the last component in the DN identifies the root entry of the directory.

Each attribute contains a type and one or more values. For example, the attribute `ou: Roles` has a type of `ou` (organizational unit) and a value of `Roles`. An *object class* is an attribute that specifies the required and optional attributes for an entry. You can find definitions of many object classes in RFC 2256.

The preceding example is represented in the LDAP Data Interchange Format (LDIF). The entry could also be represented graphically.

When searching an LDAP directory, you use a *search filter* to specify the search criteria. You can use an asterisk as a wildcard character. For example:

```
(cn=John S*)
```

Using an LDAP Server for Repository User Management

You can configure the Java CAPS Repository to use an LDAP server for user management. When a user attempts to log into the Repository, the user name and password are checked against the user name and password that are stored in the LDAP server. In addition, the list of roles for the user is retrieved from the server to authorize the user's access to various objects in the Repository.

To configure LDAP support with Java CAPS, you need to configure the LDAP server and then configure the Java CAPS Repository. See the appropriate section below to configure the LDAP server:

- "Configuring Oracle Internet Directory for the Repository" on page 7
- "Configuring Oracle Virtual Directory for the Repository" on page 6
- "Configuring Oracle Directory Server Enterprise Edition for the Repository" on page 9
- "Configuring the Active Directory Service for the Repository" on page 10
- "Configuring the OpenLDAP Directory Server for the Repository" on page 11

You configure the Repository so it can locate the LDAP server and find the appropriate information (such as the portion of the directory that contains users). For instructions, see "Configuring the Repository for LDAP Support" on page 12. If you want to encrypt communications between the Repository and the LDAP server, see "Configuring the Repository for LDAP and SSL Support" on page 16.

Managing Java CAPS Users provides basic information about Repository user management.

Configuring Oracle Virtual Directory for the Repository

Oracle Virtual Directory accesses information from multiple directories and databases, giving you a single entry point into the information stored in these directories. Oracle Virtual

Directory does not store user and group entries, so instead of configuring Oracle Virtual Directory you configure the LDAP servers to which it connects.

You can perform most administrative tasks, such as configuring the schema and managing the LDAP directory entries, through the Oracle Directory Services Manager or using a set of command-line tools. Oracle Directory Services Manager is available from Oracle Enterprise Manager Fusion Middleware Control or directly from its own URL.

The Data Browser on the Oracle Directory Services Manager lets you browse, add, and modify entries using the Data Browser . Directory entries appear in the data tree in the left panel, which you can expand to see more information.

Note – For detailed information about how to administrative tasks in Oracle Virtual Directory, see the documentation provided with Oracle Virtual Directory.

▼ To Configure LDAP Servers Connected to Oracle Virtual Directory

Perform the following general steps to create the user and roles for each LDAP directory that will connect to Java CAPS through the Oracle Virtual Directory. More complete instructions are provided for certain LDAP directories in the following sections:

- "Configuring Oracle Internet Directory for the Repository" on page 7
- "Configuring Oracle Directory Server Enterprise Edition for the Repository" on page 9
- "Configuring the Active Directory Service for the Repository" on page 10
- "Configuring the OpenLDAP Directory Server for the Repository" on page 11

1. **Create the `admin` user and the `Administrator` user under the directory where user entries are stored.**
2. **Create the roles `all`, `administration`, and `management` under the top node.**
3. **Assign the new roles you just created to the `admin` user and the `Administrator` user.**
4. **Go to "Configuring the Repository for LDAP Support" on page 12.**

Configuring Oracle Internet Directory for the Repository

Oracle Internet Directory runs as an application on an Oracle database. It includes the following main components:

- Oracle directory server
- Oracle directory replication server

- Directory administration tools, including:
  - Oracle Directory Services Manager
  - Command-line tools
  - Oracle Internet Directory pages in Oracle Enterprise Manager Fusion Middleware Control
- Oracle Internet Directory Software Developer's Kit

As with Oracle Virtual Directory, you can perform administrative tasks, such as configuring the schema and managing the LDAP directory entries, using Oracle Directory Services Manager (described in "Configuring Oracle Internet Directory for the Repository" on page 7) or a set of command line tools. Oracle Directory Services Manager is available from Oracle Enterprise Manager Fusion Middleware Control or directly from its own URL.

Note – For detailed information about how to perform the following steps, see the documentation provided with Oracle Internet Directory.

▼ To Configure Oracle Internet Directory

1. **Connect to the Oracle Directory Services Manager (either through Oracle Fusion Middleware Control or directory through its URL).**

2. **Create the `admin` user and the `Administrator` user in the directory containing the LDAP users. Assign these users the following object classes:**

   - person
   - top
   - organizationalPerson

3. **Create a new organizational unit for Java CAPS roles in your domain, and assign it a unique name (for example, `CAPSRoles`). Assign the new unit the following object classes:**

   - organizationalUnit
   - top

4. **Under the new organizational unit, create the following groups: `all`, `administration`, and `management`. Assign the groups the following object classes:**

   - organizationalRole
   - top
   - groupOfUniqueNames

5. **Add the `admin` user and the `Administrator` user as unique members of all the groups that you created.**

6 **Go to "Configuring the Repository for LDAP Support" on page 12.**

Configuring Oracle Directory Server Enterprise Edition for the Repository

Oracle Directory Server Enterprise Edition version 5.x includes the following primary components:

- Directory Server
- Administration Server
- Directory Server console

The Directory Server console enables you to perform most administrative tasks. The console contains four top-level tabs: Tasks, Configuration, Directory, and Status. The Directory tab displays the directory entries as a tree. You can browse, display, and edit all of the entries and attributes from this tab.

You can also perform administrative tasks manually by editing configuration files or by using command-line utilities.

Oracle Directory Server Enterprise Edition version 6.*x* provides the following ways for you to manage the entries in a directory:

- Directory Service Control Center (DSCC)
- Directory Editor
- `ldapmodify` and `ldapdelete` command-line utilities

DSCC is integrated into the Oracle Java Web Console. DSCC contains five top-level tabs: Common Tasks, Directory Servers, Proxy Servers, Server Groups, and Settings. To access the page where you can browse, add, and modify entries, click the Directory Servers tab, click the name of a server, and then click the Entry Management tab. The Directory Information Tree (DIT) appears on the left.

You can also use the Common Tasks tab to create a new entry or browse data.

Note – For detailed information about how to perform the following steps, see the documentation provided with Oracle Directory Server Enterprise Edition.

▼ To Configure Oracle Directory Server Enterprise Edition

1 **Create the `admin` user and the `Administrator` user under the `People` directory.**

2 **Create the roles `all`, `administration`, and `management` under the top node.**

3 **Assign the roles that you created to the `admin` user and the `Administrator` user.**

4 **Go to "Configuring the Repository for LDAP Support" on page 12.**

Configuring the Active Directory Service for the Repository

Active Directory is a key part of Windows 2003. It provides a wide variety of manageability, security, and interoperability features. The main administration tool is a snap-in called Active Directory Users and Computers.

Active Directory does not support the concept of roles. Therefore, you must simulate the Java CAPS roles in Active Directory using the concept of *groups*.

Rather than creating the groups within the `Users` directory, you create the groups in a new organizational unit called `CAPSRoles`.

Note – For detailed information about how to perform the following steps, see the documentation provided with Active Directory.

▼ To Configure the Active Directory Service

1 **Start the Active Directory Users and Computers administration tool.**

2 **Create a new organizational unit for Java CAPS roles:**

   a. **Right-click the root node and select New > Organizational Unit.**
      The New Object - Organization Unit dialog box appears.

   b. **In the Name field, enter a value (for example, `CAPSRoles`).**

   c. **Click OK.**

3 **Under the organizational unit, create the following groups: `all`, `administration`, and `management`. To create a group, you right-click the organizational unit and select New > Group. Use the default values for Group scope and Group type.**
   After you add the groups, they appear under the organizational unit.

4 **Add the `admin` user and the `Administrator` user as members of all the groups that you created by double-clicking each group and selecting `admin` and `Administrator` from the dialog box.**

5 **Go to "Configuring the Repository for LDAP Support" on page 12.**

Configuring the OpenLDAP Directory Server for the Repository

The OpenLDAP Project provides an open source implementation of the LDAP protocol. The LDAP server runs as a standalone daemon called `slapd`. The main configuration file is called `slapd.conf`. This file contains global information specific to the database and the back end. You can use various approaches to add entries to the database, such as using the `slapadd` program. To search the database, use the `ldapsearch` program.

For more information, see `http://www.openldap.org`.

Note – For detailed information about how to perform the following steps, see the documentation provided with OpenLDAP Directory Server.

▼ To Configure the OpenLDAP Directory Server

1. **Create the `admin` user and the `Administrator` user under the node where the users are located.**

2. **If you do not have a node for roles in your schema, then create a node for the Java CAPS-specific roles that you will create in the following step. For example:**

```
dn: ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: organizationalUnit
ou: CAPSRoles
```

3. **Create the roles `all`, `administration`, and `management` under the node where the roles are located. Add the `admin` user and the `Administrator` user as unique members of each role. For example:**

```
dn: cn=all, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com

dn: cn=administration, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: administration
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com

dn: cn=management, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: management
```

```
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com
```

4 **Add other users to one or more roles, as necessary. For example:**

```
dn: cn=all, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: all
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com
uniqueMember: uid=userA, ou=People, dc=oracle, dc=com
uniqueMember: uid=userB, ou=People, dc=oracle, dc=com

dn: cn=administration, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: administration
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com
uniqueMember: uid=userB, ou=People, dc=oracle, dc=com

dn: cn=management, ou=CAPSRoles, dc=oracle, dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: management
ou: CAPSRoles
uniqueMember: uid=admin, ou=People, dc=oracle, dc=com
uniqueMember: uid=Administrator, ou=People, dc=oracle, dc=com
```

5 **Go to "Configuring the Repository for LDAP Support" on page 12.**

Configuring the Repository for LDAP Support

To use an LDAP server for Repository user management, you must add a `<Realm>` element to the Repository's `server.xml` file, which is located in the *JavaCAPS-install-dir*`/repository/repository/server/conf` directory. The `server.xml` file contains a default `<Realm>` element that specifies a flat file implementation of the user database. The flat file implementation uses the `tomcat-users.xml` file in the *JavaCAPS-install-dir*`/repository/repository/data/files` directory.

The following table describes the attributes used by the LDAP versions of the `<Realm>` element. For a detailed description of all the possible attributes, see the Tomcat documentation for the `org.apache.catalina.realm.JNDIRealm` class.

Attribute	Description
className	Always use the following value: `org.apache.catalina.realm.JNDIRealm`

Attribute	Description
connectionURL	Identifies the location of the LDAP server. Includes the LDAP server name and the port that the LDAP server listens on for requests.
roleBase	The base entry for the role search. If this attribute is not specified, then the search base is the top-level directory context.
roleName	The attribute in a role entry containing the name of the role.
roleSearch	The LDAP search filter for selecting role entries. It optionally includes pattern replacements `{0}` for the Distinguished Name and/or `{1}` for the user name of the authenticated user. In certain cases of an authenticated user (for example, Administrator), option `{0}` should be selected.
roleSubtree	By default, the Roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`.
userBase	The entry that is the base of the subtree containing users. If this attribute is not specified, then the search base is the top-level context.
userPattern	A pattern for the Distinguished Name (DN) of the user's directory entry, following the syntax supported by the `java.text.MessageFormat` class with `{0}` indicating where the actual user name should be inserted.
userRoleName	The name of an attribute in the user's directory entry containing zero or more values for the names of roles assigned to this user. In addition, you can use the `roleName` attribute to specify the name of an attribute to be retrieved from individual role entries found by searching the directory. If `userRoleName` is not specified, then all roles for a user derive from the role search.
userRoleNamePattern	A pattern for the Distinguished Name (DN) of the role's directory entry, following the syntax supported by the `java.text.MessageFormat` class with `{0}` indicating the actual role name. This pattern is used to parse the DN to get the actual role name for authorization purposes in Java CAPS, where the actual user name should be inserted.
userSearch	The LDAP search filter to use for selecting the user entry after substituting the user name in `{0}`.
userSubtree	By default, the Users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`.

▼ To Configure the Repository

1. **Open the `server.xml` file in the `JavaCAPS-install-dir/repository/repository/server/conf` directory.**

2. **Remove or comment out the default `<Realm>` element.**

3 If you are using Oracle Internet Directory or Oracle Virtual Directory, add the following `<Realm>` element inside the `<Engine>` tag. Change the values shown below as necessary. The preceding table describes the attributes.

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:3060"
    connectionName="cn=oracleadmin"
    connectionPassword="OpCT/AcQGL/ch+GN460Zcg="
    userBase="cn=People,dc=oracle,dc=com"
    userSearch="(cn={0})"
    userSubtree="true"
    roleBase="ou=CAPSRoles,dc=sun,dc=com"
    roleName="cn"
    roleSearch="(uniqueMember={0})"
    roleSubtree="true"
/>
```

Note – For the `connectionName` property, enter the DN of the administrator user. The value of the `connectionPassword` property must be encrypted. You can use the encrypt utility provided with Java CAPS, located in *JavaCAPS_Home*`\repository\repository\util`. This utility uses the following syntax:

`encrypt` *passsword*

Where *password* is the unencrypted password for the user. The utility will display the encrypted version of the password.

---

4 If you are using Oracle Directory Server Enterprise Edition, add the following `<Realm>` element inside the `<Engine>` tag. Change the values shown below as necessary. The preceding table describes the attributes.

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:489"
    userBase="cn=People,dc=oracle,dc=com"
    userSearch="(uid={0})"
    userSubtree="true"
    userRoleName="nsroledn"
    userRoleNamePattern="cn={0},dc=oracle,dc=com"
    roleSubtree="true"
/>
```

5 If you are using Active Directory, add the following `<Realm>` element inside the `<Engine>` tag. Change the values shown below as necessary. The preceding table describes the attributes.

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:389"
    userBase="cn=Users,dc=oracle,dc=com"
    userSearch="(cn={0})"
    userSubtree="true"
    roleBase="ou=CAPSRoles,dc=oracle,dc=com"
    roleName="cn"
    roleSearch="(member={0})"
    roleSubtree="true"
/>
```

6. **If you are using OpenLDAP Directory Server, add the following `<Realm>` element inside the `<Engine>` tag. Change the values shown below as necessary. The preceding table describes the attributes.**

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldap://localhost:389"
    userBase="ou=People,dc=oracle,dc=com"
    userSearch="(uid={0})"
    userSubtree="true"
    roleBase="ou=CAPSRoles,dc=oracle,dc=com"
    roleName="cn"
    roleSearch="(uniquemember={0})"
    roleSubtree="true"
/>
```

7. **If your LDAP server is not configured for anonymous read access, add the `connectionName` and `connectionPassword` attributes to the `<Realm>` element. Set the first attribute to the DN of the `Administrator` user. Set the second attribute to the user's encrypted password. Refer to the following examples.**

   Oracle Directory Server Enterprise Edition:

```
connectionName="cn=Directory Manager"
connectionPassword="E451KDVb0OPcH+GN46OZcg=="
```

   Active Directory:

```
connectionName="Administrator@oracle.com"
connectionPassword="geEiVIbtO+DcH+GN46OZcg=="
```

   OpenLDAP Directory Server:

```
connectionName="cn=Manager,dc=oracle,dc=com"
connectionPassword="l/ZRt1cfNKc="
```

   To encrypt the password, use the `encrypt` utility in the `JavaCAPS-install-dir/repository/repository/util` directory. The file extension of the utility depends on your platform. This utility takes the unencrypted password as an argument. For example:

```
C:\JavaCAPS6\repository\repository\util>encrypt mypwd
LCUApSkYpuE
```

8. **Save and close the `server.xml` file.**

9. **Start the LDAP server.**

10. **Shut down and restart the Repository.**

Configuring the Repository for LDAP and SSL Support

By default, communications between the Repository and the LDAP server are unencrypted. To encrypt communications between the Repository and the LDAP server, make the following additions and modifications to the procedures described earlier in this topic.

Configuring SSL on the LDAP Server

Ensure that the LDAP server is configured to use the Secure Sockets Layer (SSL). For detailed instructions, see the documentation provided with the LDAP server. In preparation for the next step, export the LDAP server's certificate to a file.

Importing the LDAP Server's Certificate

You must add the LDAP server's certificate to the Repository's list of trusted certificates. The list is located in a file called `cacerts`. In the following procedure, you use the `keytool` program. This program is included with the Java SDK.

▼ To Import the LDAP Server's Certificate

1. **Navigate to the `JDK-install-dir/jre/bin` directory.**

   Use the JDK that was specified during the installation of the Repository.

2. **Run the following command:**

   ```
   keytool -import -trustcacerts -alias alias -file certificate_filename
   -keystore cacerts_filename
   ```

   For the `-alias` option, you can assign any value.

   For the `-file` option, specify the fully qualified name of the LDAP server's certificate. For example:

   ```
   C:\mycertificate.cer
   ```

   For the `-keystore` option, specify the fully qualified name of the `cacerts` file. The `cacerts` file is located in the `JDK-install-dir/jre/lib/security` directory. For example:

   ```
   C:\Java\jdk1.6.0_06\jre\lib\security\cacerts
   ```

3. **When prompted, enter the keystore password. The default password is `changeit`.**

4. **When prompted to trust this certificate, enter `yes`.**

   The following message appears:

   ```
   Certificate was added to keystore
   ```

Modifying the LDAP Server URL

To use the Repository with LDAP and SSL, you need to modify the `Realm` element you created when you performed the steps described in "Configuring the Repository for LDAP Support" on page 12.

▼ To Modify the LDAP Server URL

1. **Navigate to *JavaCAPS_Home*`\repository\repository\server\conf`.**
2. **Open `server.xml` in a text editor.**
3. **In the `Realm` element you created for the LDAP server, update the `connectURL` property by setting the protocol to `ldaps` and setting the port number to the port number that the LDAP server listens on for SSL requests.**

   Typically, this number is 636. For example:

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
    connectionURL="ldaps://myldapserver:636"
    ...
```

Using an LDAP Server for Oracle Java CAPS JMS IQ Manager User Management

You can configure a Oracle Java CAPS JMS IQ Manager to use an LDAP server for user management. A *realm* is a collection of users, groups, and roles that are used in enforcing security policies. The JMS IQ Manager supports multiple LDAP realms running at the same time.

When you perform the following steps, access to the JMS IQ Manager is granted only when the connection has a valid user name and password.

For a list of supported LDAP servers, see "Java CAPS 6.3 Components and Supported External Systems" in *Planning for Oracle Java CAPS 6.3 Installation* . For basic information about Oracle Java CAPS JMS IQ Manager user management, see *Managing Java CAPS Users*.

Configuring the LDAP Server

In the following procedure, you create users and roles in the LDAP server.

▼ To Configure the LDAP server

1. **Create one or more JMS IQ Manager users.**
2. **Create one or more of the following roles:**

Role	Description
application	Enables clients to access the JMS IQ Manager.
asadmin	Enables use of the JMS control utility (`stcmsctrlutil`) or Enterprise Manager, and enables clients to access the JMS IQ Manager.

3. **Assign the roles to your users as needed.**

Configuring the Oracle Java CAPS JMS IQ Manager

You must configure the JMS IQ Manager so it can locate the LDAP server and find the appropriate information. You can enable more than one LDAP server and you can specify the default realm.

▼ To Configure the Oracle Java CAPS JMS IQ Manager

1. **If the GlassFish server is not running, start it before proceeding.**
2. **Log in to the Configuration Agent. The format of the URL is `http://hostname:port-number/configagent`. Set the hostname to the TCP/IP host name of the computer where the application server is installed. Set the port number to the administration port number of the application server. For example:**

   ```
   http://localhost:4848/configagent
   ```

3. **In the left pane, click the JMS IQ Manager node (for example, `IQ_Manager_18007`).**
4. **Click the Access Control tab.**
5. **Ensure that the check box to the right of the Require Authentication label is selected.**
6. **If you want to change the default realm, select a new realm from the Default Realm drop-down list.**
7. **To disable the file realm, deselect the check box to the right of Enable File Realm.**

Note – Disable file realm when using Oracle Internet Directory or Oracle Virtual Directory.

8. **To enable Oracle Directory Server Enterprise Edition, select the check box to the right of Enable Sun Java System Directory Server and click Show Properties. Modify the values for the properties as described in Table 1.**

   The default values are intended to match the standard schema of Oracle Directory Server Enterprise Edition.

9. **To enable Active Directory, select the check box to the right of Enable Microsoft Active Directory Server and click Show Properties. Modify the values for the properties as described in Table 2.**

   The default values are intended to match the standard schema of Active Directory.

10. **To enable OpenLDAP Directory Server, select the check box to the right of Enable Generic LDAP Server and click Show Properties. Modify the values for the properties as described in Table 3.**

11. **To enable Oracle Internet Directory, select the check box to the right of Enable Oracle Internet Directory Server and click Show Properties. Enter values for the properties as described in Table 4.**

    The default values are intended to match the standard schema of Oracle Internet Directory.

12. **To enable Oracle Virtual Directory, select the check box to the right of Enable Oracle Virtual Directory Server and click Show Properties. Enter values for the properties as described in Table 5.**

    The following table describes the properties that appear. The default values are intended to match the standard schema of Oracle Directory Server Enterprise Edition. Review the default value for each property. If necessary, modify the default value.

13. **Click Save.**

Access Control LDAP Server Properties

The following tables describe the access control properties that appear for each LDAP server:

- Oracle Directory Server Enterprise Edition Access Control Properties
- Microsoft Active Directory Server Access Control Properties
- OpenLDAP Directory Server Access Control Properties
- Oracle Internet Directory Access Control Properties
- Oracle Virtual Directory Access Control Properties

The following table lists the Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server) properties on the Access Control page of the Configuration Agent.

TABLE 1 Oracle Directory Server Enterprise Edition Access Control Properties

Property	Description
Naming Provider URL	The URL of the Java Naming and Directory Interface (JNDI) service provider. The default value is `ldap://IP_address:589`.
Naming Initial Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations. The default value is `com.sun.jndi.ldap.LdapCtxFactory`.
Naming Security Authentication	The security level to use in JNDI naming operations. The default value is `simple`.
Naming Security Principal	The security principal used for connecting to the LDAP server.
Naming Security Credentials	The password of the naming security principal. The default value is `STC`. The value is encrypted when you save and then view it again.
Group DN Attribute Name in Group	The name of the Distinguished Name attribute in group entries. The default value is `entrydn`.
Group Name Field in Group DN	The name of the group name field in group Distinguished Names. The default value is `cn`.
Groups of User Filter Under Groups Parent DN	The LDAP search filter used to retrieve all of a user's groups. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `uniquemember={1}`.
Groups Parent DN	The parent Distinguished Name of the group entries. In other words, this property specifies the root entry of the groups portion of the LDAP directory.
Role Name Attribute Name in User	The name of the role name attribute in user entries. The default value is `nsroledn`.

TABLE 1 Oracle Directory Server Enterprise Edition Access Control Properties *(Continued)*

Property	Description
Role Name Field in Role DN	The name of the role name field in role Distinguished Names. The default value is `cn`.
Roles Parent DN	The parent Distinguished Name of the role entries. In other words, this property specifies the root entry of the roles portion of the LDAP directory.
Search Groups Sub Tree	By default, the groups portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Roles Sub Tree	By default, the roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Users Sub Tree	By default, the users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
User DN Attribute Name in User	The name of the Distinguished Name attribute in user entries. The default value is `entrydn`.
User ID Attribute Name in User	The name of the user ID attribute in user entries. The default value is `uid`.
Users Parent DN	The parent Distinguished Name of the user entries. In other words, this property specifies the root entry of the users portion of the LDAP directory.

The following table lists the Microsoft Active Directory Server properties on the Access Control page of the Configuration Agent.

TABLE 2 Microsoft Active Directory Server Access Control Properties

Property	Description
Naming Provider URL	The URL of the Java Naming and Directory Interface (JNDI) service provider. The default value is `ldap://IP_address:389`.
Naming Initial Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations. The default value is `com.sun.jndi.ldap.LdapCtxFactory`.
Naming Security Authentication	The security level to use in JNDI naming operations. The default value is `simple`.
Naming Security Principal	The security principal used for connecting to the LDAP server.
Naming Security Credentials	The password of the naming security principal. The default value is `STC`. The value is encrypted when you save and then view it again.
Users Parent DN	The parent Distinguished Name of the user entries. In other words, this property specifies the root entry of the users portion of the LDAP directory.
User DN Attribute Name in User	The name of the Distinguished Name attribute in user entries. The default value is `distinguishedName`.
User ID Attribute Name in User	The name of the user ID (that is, the login ID) attribute in user entries. The default value is `sAMAccountName`.
Roles Parent DN	The parent Distinguished Name of the role entries. In other words, this property specifies the root entry of the roles portion of the LDAP directory.
Role DN Attribute Name in Role	The name of the Distinguished Name attribute in role entries. The default value is `cn`.

TABLE 2 Microsoft Active Directory Server Access Control Properties *(Continued)*

Property	Description
Roles of User Filter Under Roles Parent DN	The LDAP search filter used to retrieve all of a user's roles. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `(&(member={1})(objectclass=group))`.
Groups Parent DN	The parent Distinguished Name of the group entries. In other words, this property specifies the root entry of the groups portion of the LDAP directory.
Group DN Attribute Name in Group	The name of the Distinguished Name attribute in group entries. The default value is `distinguishedName`.
Group Name Field in Group DN	The name of the group name field in group Distinguished Names. The default value is `cn`.
Groups of User Filter Under Groups Parent DN	The LDAP search filter used to retrieve all of a user's groups. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `(&(member={1})(objectclass=group))`.
Search Groups Sub Tree	By default, the groups portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Users Sub Tree	By default, the users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.

TABLE 2 Microsoft Active Directory Server Access Control Properties *(Continued)*

Property	Description
Search Roles Sub Tree	By default, the roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.

The following table lists the OpenLDAP Directory Server properties on the Access Control page of the Configuration Agent.

TABLE 3 OpenLDAP Directory Server Access Control Properties

Property	Description
Naming Provider URL	The URL of the Java Naming and Directory Interface (JNDI) service provider. The default value is `ldap://IP_address:489`.
Naming Initial Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations. The default value is `com.sun.jndi.ldap.LdapCtxFactory`.
Naming Security Authentication	The security level to use in JNDI naming operations. The default value is `simple`.
Users Parent DN	The parent Distinguished Name of the user entries. In other words, this property specifies the root entry of the users portion of the LDAP directory.
User ID Attribute Name in User	The name of the user ID attribute in user entries. The default value is `uid`.
Roles Parent DN	The parent Distinguished Name of the role entries. In other words, this property specifies the root entry of the roles portion of the LDAP directory.
Role Name Attribute Name in Role	The name of the role name attribute in user entries. The default value is `cn`.

TABLE 3 OpenLDAP Directory Server Access Control Properties *(Continued)*

Property	Description
Roles of User Filter Under Roles Parent DN	The LDAP search filter used to retrieve all of a user's roles. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `uniquemember={1}`.
Group Name Field in Group DN	The name of the group name field in group Distinguished Names. The default value is `cn`.
Groups Parent DN	The parent Distinguished Name of the group entries. In other words, this property specifies the root entry of the groups portion of the LDAP directory.
Groups of User Filter Under Groups Parent DN	The LDAP search filter used to retrieve all of a user's groups. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `uniquemember={1}`.
Search Groups Sub Tree	By default, the groups portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Users Sub Tree	By default, the users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Roles Sub Tree	By default, the roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.

The following table lists the Oracle Internet Directory properties on the Access Control page of the Configuration Agent.

TABLE 4 Oracle Internet Directory Access Control Properties

Property	Description
Naming Provider URL	The URL of the Java Naming and Directory Interface (JNDI) service provider. The default value is `ldap://127.0.0.1:3060`.
Naming Initial Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations. The default value is `com.sun.jndi.ldap.LdapCtxFactory`.
Naming Security Authentication	The security level to use in JNDI naming operations. The default value is `simple`.
Naming Security Principal	The security principal to use for connecting to the LDAP server. The default value is `cn=orcladmin`.
Naming Security Credentials	The password of the naming security principal. The default value is `welcome1`. The value is encrypted when you save and then view it again.
Users Parent DN	The parent Distinguished Name of the user entries. This property specifies the root entry of the users portion of the LDAP directory. The default value is `cn=People,dc=sun,dc=com`.
User ID Attribute Name in User	The name of the user ID attribute in user entries. The default value is `cn`.
Roles Parent DN	The parent Distinguished Name of the role entries. This property specifies the root entry of the roles portion of the LDAP directory. The default value is `ou=capsroles,dc=sun,dc=com`.
Role Name Attribute Name in User	The name of the role name attribute in user entries. The default value is `cn`.

TABLE 4 Oracle Internet Directory Access Control Properties *(Continued)*

Property	Description
Roles of User Filter Under Roles Parent DN	The LDAP search filter used to retrieve all of a user's roles. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `(uniqueMember={1})`.
Search Roles Sub Tree	By default, the roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Users Sub Tree	By default, the users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.

The following table lists the Oracle Virtual Directory properties on the Access Control page of the Configuration Agent.

TABLE 5 Oracle Virtual Directory Access Control Properties

Property	Description
Naming Provider URL	The URL of the Java Naming and Directory Interface (JNDI) service provider. The default value is `ldap://127.0.0.1:6501`.
Naming Initial Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations. The default value is `com.sun.jndi.ldap.LdapCtxFactory`.
Naming Security Authentication	The security level to use in JNDI naming operations. The default value is `simple`.
Naming Security Principal	The security principal to use for connecting to the LDAP server. The default value is `cn=orcladmin`.

TABLE 5 Oracle Virtual Directory Access Control Properties *(Continued)*

Property	Description
Naming Security Credentials	The password of the naming security principal. The default value is `welcome1`. The value is encrypted when you save and then view it again.
Users Parent DN	The parent Distinguished Name of the user entries. This property specifies the root entry of the users portion of the LDAP directory. The default value is `cn=People,dc=sun,dc=com`.
User ID Attribute Name in User	The name of the user ID attribute in user entries. The default value is `cn`.
Roles Parent DN	The parent Distinguished Name of the role entries. This property specifies the root entry of the roles portion of the LDAP directory. The default value is `ou=capsroles,dc=sun,dc=com`.
Role Name Attribute Name in User	The name of the role name attribute in user entries. The default value is `cn`.
Roles of User Filter Under Roles Parent DN	The LDAP search filter used to retrieve all of a user's roles. This property follows the syntax supported by the `java.text.MessageFormat` class with `{1}` indicating where the user's Distinguished Name should be inserted. The default value is `(uniqueMember={1})`.
Search Roles Sub Tree	By default, the roles portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.
Search Users Sub Tree	By default, the users portion of the LDAP directory is searched only one level below the root entry. To enable searches of the entire subtree, set the value to `true`. The default value is `false`.

Using an LDAP Server for Enterprise Manager User Management

You can configure Enterprise Manager to use an LDAP server for user management. This is a two-step process. First, you configure the LDAP server, and then you configure the Enterprise Manager server so it can locate the LDAP server and find the appropriate information (for example, the portion of the directory that contains users).

See the following topics for instructions on how to configure different types of LDAP directories:

- "Configuring Oracle Virtual Directory for Enterprise Manager" on page 29
- "Configuring Oracle Internet Directory for Enterprise Manager" on page 30
- "Configuring Oracle Directory Server Enterprise Edition for Enterprise Manager" on page 31
- "Configuring Microsoft Active Directory Service for Enterprise Manager" on page 31
- "Configuring the OpenLDAP Directory Server for Enterprise Manager" on page 32

Once you configure the LDAP directory, you need to configure the Enterprise Manager, as described in "Configuring the Enterprise Manager Server" on page 33.

Managing Java CAPS Users provides basic information about Enterprise Manager user management.

Configuring Oracle Virtual Directory for Enterprise Manager

Oracle Virtual Directory accesses information from multiple directories and databases, giving you a single entry point into the information stored in these directories. For more information about Oracle Virtual Directory, see "Configuring Oracle Virtual Directory for the Repository" on page 6.

▼ To Configure LDAP Servers Connected to Oracle Virtual Directory

Perform the following general steps to create the user and roles for each LDAP directory that will connect to the Enterprise Manager through the Oracle Virtual Directory. More complete instructions are provided for specific LDAP directories in the following sections:

- "Configuring Oracle Internet Directory for Enterprise Manager" on page 30
- "Configuring Oracle Directory Server Enterprise Edition for Enterprise Manager" on page 31
- "Configuring Microsoft Active Directory Service for Enterprise Manager" on page 31
- "Configuring the OpenLDAP Directory Server for Enterprise Manager" on page 32

1. **If you have not done so already, create the `admin` user and the `Administrator` user under the `People` directory.**

2. **Create the following roles under the top node:**

   - Deployment
   - User Management
   - Read-Only Monitor
   - Controlling Monitor
   - JMS Read-Only Monitor
   - JMS Read-Write Monitor
   - Manager

3. **Assign the roles that you created to the `admin` user and the `Administrator` user.**

Configuring Oracle Internet Directory for Enterprise Manager

Oracle Internet Directory runs as an application on an Oracle database. For more information about Oracle Internet Directory, see "Configuring Oracle Internet Directory for the Repository" on page 7.

▼ To Configure Oracle Internet Directory

You only need to perform steps 2 and 3 (creating the Java CAPS users and organizational unit for roles) if you did not already create them when configuring LDAP for the Repository. For more information, see "Configuring Oracle Internet Directory for the Repository" on page 7.

1. **Connect to the Oracle Directory Services Manager (either through Oracle Fusion Middleware Control or directory through its URL).**

2. **If you have not done so already, create the `admin` user and the `Administrator` user in the directory containing the LDAP users.**

3. **If you have not done so already, create a new organizational unit for Java CAPS roles in your domain, and assign it a unique name (for example, `CAPSRoles`).**

4. **Under the new organizational unit, create the following groups:**

   - Deployment
   - User Management
   - Read-Only Monitor
   - Controlling Monitor
   - JMS Read-Only Monitor

- JMS Read-Write Monitor
- Manager

5 **Add the `admin` user and the `Administrator` user as unique members of all the groups that you created.**

6 **Go to "Configuring the Enterprise Manager Server" on page 33.**

Configuring Oracle Directory Server Enterprise Edition for Enterprise Manager

Oracle Directory Server Enterprise Edition provides a console for you to perform administrative tasks. For more information about Oracle Directory Server Enterprise Edition, see "Configuring Oracle Directory Server Enterprise Edition for the Repository" on page 9.

▼ To Configure the Oracle Directory Server Enterprise Edition

1 **If you have not done so already, create the `admin` user and the `Administrator` user under the `People` directory.**

2 **Create the following roles under the top node:**

- Deployment
- User Management
- Read-Only Monitor
- Controlling Monitor
- JMS Read-Only Monitor
- JMS Read-Write Monitor
- Manager

3 **Assign the roles that you created to the `admin` user and the `Administrator` user.**

4 **Go to "Configuring the Enterprise Manager Server" on page 33.**

Configuring Microsoft Active Directory Service for Enterprise Manager

Active Directory is a key part of Windows 2000. It provides a wide variety of manageability, security, and interoperability features. The main administration tool is a snap-in called Active Directory Users and Computers.

Active Directory does not support the concept of roles. Therefore, you must simulate the Enterprise Manager roles in Active Directory using the concept of *groups*.

Note – For detailed information about how to perform the following steps, see the documentation provided with Active Directory.

▼ To Configure the Active Directory Service

1. **Start the Active Directory Users and Computers administration tool.**
2. **Right-click the root node and select New > Organizational Unit.**

   The New Object - Organization Unit dialog box appears.
3. **In the Name field, enter a value (for example, `EntMgrRoles`).**
4. **Click OK.**
5. **Under the organizational unit, create the following groups:**

   - Deployment
   - User Management
   - Read-Only Monitor
   - Controlling Monitor
   - JMS Read-Only Monitor
   - JMS Read-Write Monitor
   - Manager

   After you add the groups, they appear under the organizational unit.
6. **Add the `admin` user and the `Administrator` user as members of all the groups that you created by double-clicking each group and selecting `admin` and `Administrator` from the dialog box.**
7. **Go to "Configuring the Enterprise Manager Server" on page 33.**

Configuring the OpenLDAP Directory Server for Enterprise Manager

The OpenLDAP Project provides an open source implementation of the LDAP protocol. The LDAP server runs as a standalone daemon called `slapd`. The main configuration file is called `slapd.conf`. This file contains global information that is specific to the database and back end. You can use various approaches to add entries to the database, such as using the `slapadd` program. To search the database, use the `ldapsearch` program.

For more information, see `http://www.openldap.org`.

Note – For detailed information about how to perform the following steps, see the documentation provided with OpenLDAP Directory Server.

▼ To Configure the OpenLDAP Directory Server

1. **Create the `admin` user and the `Administrator` user under the node where the users are located.**
2. **If you do not have a node for roles in your schema, then create a node for the Enterprise Manager roles that you will create in the following step.**
3. **Create the following roles under the node where the roles are located:**
   - Deployment
   - User Management
   - Read-Only Monitor
   - Controlling Monitor
   - JMS Read-Only Monitor
   - JMS Read-Write Monitor
   - Manager
4. **Add the `admin` user and the `Administrator` user as unique members of each role.**
5. **Add other users to one or more roles, as necessary.**
6. **Go to "Configuring the Enterprise Manager Server" on page 33.**

Configuring the Enterprise Manager Server

Once you have configured the LDAP server, you configure the Enterprise Manager server so that it can locate the LDAP server and find the appropriate information.

You must edit the following Enterprise Manager files: `web.xml` and `ldap.properties`.

▼ To Configure the Enterprise Manager Server

1. **Shut down the server component of Enterprise Manager.**
2. **Open the `web.xml` file in the `JavaCAPS-install-dir/emanager/server/webapps/sentinel/WEB-INF` directory.**

3 Locate the following lines:

```
<param-name>com.stc.emanager.sentinel.authHandler</param-name>
<param-value>com.stc.cas.auth.provider.tomcat.TomcatPasswordHandler</param-value>
```

4 Change the parameter value to:

```
<param-value>com.stc.cas.auth.provider.ldap.LDAPHandler</param-value>
```

5 Save the `web.xml file`.

6 Open the `ldap.properties` file in the `JavaCAPS-install-dir/emanager/server/webapps/sentinel/WEB-INF/classes` directory.

7 The following table describes all of the properties that appear in the `ldap.properties` file. Edit the properties in the section for your LDAP server, and ensure that the properties are not commented out.

Property	Description
com.stc.sentinel.auth.ldap.serverType	The type of LDAP server.
com.stc.sentinel.auth.ldap.serverUrl	The URL of the LDAP server.
com.stc.sentinel.auth.ldap.searchFilter	The name of the user ID attribute in user entries.
com.stc.sentinel.auth.ldap.searchBase	The root entry of the portion of the LDAP directory where Enterprise Manager will search for users.
com.stc.sentinel.auth.ldap.searchScope	This property is not currently used.
com.stc.sentinel.auth.ldap.bindDN	The security principal used for connecting to the LDAP server.
com.stc.sentinel.auth.ldap.bindPassword	The password of the security principal.
com.stc.sentinel.auth.ldap.referral	The LDAP referral policy. The default value is `follow`, which indicates that LDAP referrals will be automatically followed. Note that referrals must be enabled in the LDAP server. The other valid values are `throw` (for referral exceptions) and `ignore`. This property is optional. This property is not included for Oracle Directory Server Enterprise Edition.
com.stc.sentinel.auth.ldap.roleAttribute	The name of the role name attribute in user entries.

Property	Description
com.stc.sentinel.auth.ldap.roleBaseDN	The root entry of the portion of the LDAP directory where Enterprise Manager will search for roles. This property appears only in the OpenLDAP set of properties.
com.stc.sentinel.auth.ldap.rolePattern	Enables you to configure pattern matching for role names. You can place the Enterprise Manager users in a separate line of business from other users in the LDAP directory. This property appears only in the Active Directory set of properties.

8 **Save the `ldap.properties` file.**

9 **Start the server component of Enterprise Manager.**

Configuring Enterprise Manager for LDAP and SSL Support

By default, communications between Enterprise Manager and the LDAP server are unencrypted. To encrypt communications, make the following additions and modifications to the procedures described earlier in this topic.

Configuring SSL on the LDAP Server

Ensure that the LDAP server is configured to use the Secure Sockets Layer (SSL). For instructions, see the documentation provided with the LDAP server. In preparation for the next step, export the LDAP server's certificate to a file.

Importing the LDAP Server's Certificate

You must add the LDAP server's certificate to the Enterprise Manager's list of trusted certificates. The list is located in a file called `cacerts`, located in the `JDK-install-dir/jre/lib/security` directory. In the following procedure, you use the `keytool` program. This program is included with the Java SDK.

▼ To Import the LDAP Server's Certificate

1 **Navigate to the *Java_Home*`\jre\lib\security` directory.**

Use the JDK that was specified during Java CAPS installation.

2. **Run the following command:**

   `keytool -import -trustcacerts -alias` *alias_name* `-file` *certificate_filename* `-keystore` *cacerts_filename*

   - For the `-alias` option, assign any value.
   - For the `-file` option, specify the fully qualified name of the LDAP server's certificate. For example:

     ```
     C:\mycertificate.cer
     ```

   - For the `-keystore` option, specify the fully qualified name of the `cacerts` file.

3. **When prompted, enter the keystore password. The default password is `changeit`.**

4. **When prompted to trust this certificate, enter `yes`.**

   The following message appears:

   ```
   Certificate was added to keystore
   ```

Modifying the LDAP Server URL

When you configured the LDAP properties for Enterprise Manager, as described in "Configuring the Enterprise Manager Server" on page 33, you specified the LDAP server URL. When using the SSL protocol, you need to modify that URL as described below.

▼ To Modify the LDAP Server URL

1. **Navigate to *JavaCAPS_Home*`\emanager\server\webapps\sentinel\WEB-INF\classes`.**

2. **Open `ldap.properties` in a text editor.**

3. **In the `com.stc.sentinel.auth.ldap.serverUrl` property, set the protocol to `ldaps` and set the port number to the port number that the LDAP server listens on for SSL requests.**

   Typically, this number is 636. For example:

   ```
   com.stc.sentinel.auth.ldap.serverUrl=ldaps://MyLDAPServer:636
   ```

4. **Save and close the file.**

Specifying an Application Configuration Property Dynamically

You can specify application configuration properties using either a static approach or a dynamic approach. Using the static approach, you specify a property value at design time in the NetBeans IDE. The property value is included in the application file. If the value needs to be

changed after deployment, then you must change the value in the NetBeans IDE, rebuild the application file, and redeploy the application file.

Using the dynamic approach, you specify an LDAP URL at design time. The URL must point to an attribute in an LDAP server. When you deploy the application file, the actual value is retrieved from the LDAP server. You can change the value in the LDAP server after deployment without performing the steps of the static approach. However, you do need to disable and then re-enable the application file in order for the change to take effect.

You can use this feature for properties that accept string values (including passwords), numeric values, or boolean values.

Note – Another approach to updating property values does not require the use of LDAP. In the `asadmin` tool, run the `extract-caps-application-configuration` command. The configuration properties of the specified application file are extracted to a properties file. Update the value of one or more properties, and then run the `import-caps-configuration` command. Restart the application.

Enabling the Application Server to Access the LDAP Server

In this task, you edit properties that specify how the application server can access the LDAP server.

▼ To Enable the Application Server to Access the LDAP Server

1. **Start the `asadmin` tool included with GlassFish Application Server.**

2. **Run the `export-caps-ldap-configuration` command. You must specify the directory where you want to store the `LDAP.properties` file.**

   ```
   asadmin> export-caps-ldap-configuration --capsconfigdir c:\temp
   ```

   The `LDAP.properties` file is generated.

3. **Using a text editor, open the `LDAP.properties` file.**

4. **Set values for the following properties, which specify how to access the LDAP server.**

   - `host`
   - `port`
   - `sslport`
   - `password`
   - `loginDN`

The `ldapVersion` property is optional. You can set this property to any numeric value.

5. **Save the `LDAP.properties` file.**

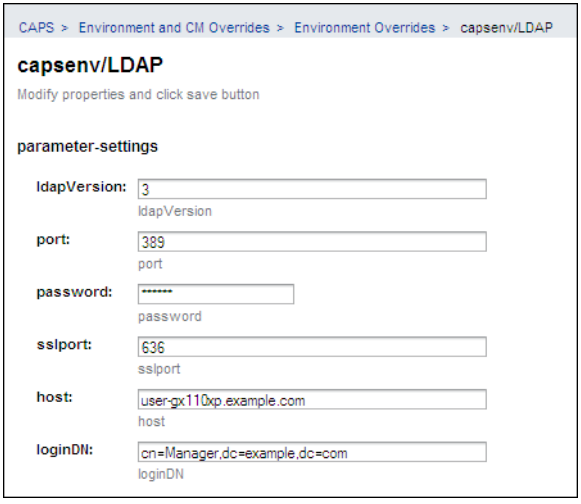
6. **Run the `import-caps-configuration` command. You must specify the directory that contains the `LDAP.properties` file.**

   ```
   asadmin> import-caps-configuration c:\temp
   ```

7. **Start the Admin Console included with GlassFish Application Server.**

8. **In the left pane, expand the `CAPS` node, the `Environment and CM Overrides` node, and the `Environment Overrides` node. Select the `capsenv/LDAP` node.**

   The property fields appear in the right pane. You can now update the properties from the Admin Console. Or you can update the `LDAP.properties` file and run the `import-caps-configuration` command again.



Specifying an LDAP URL for a Property

Here are two examples of LDAP URLs that might be used in Java CAPS:

```
ldap://uid=BatchFTP_TargetFileName,ou=Batch_Adapter,dc=Adapters,dc=oracle,dc=com?cn
ldap://uid=BatchFTP_Password,ou=Batch_Adapter,dc=Adapters,dc=oracle,dc=com?cn
```

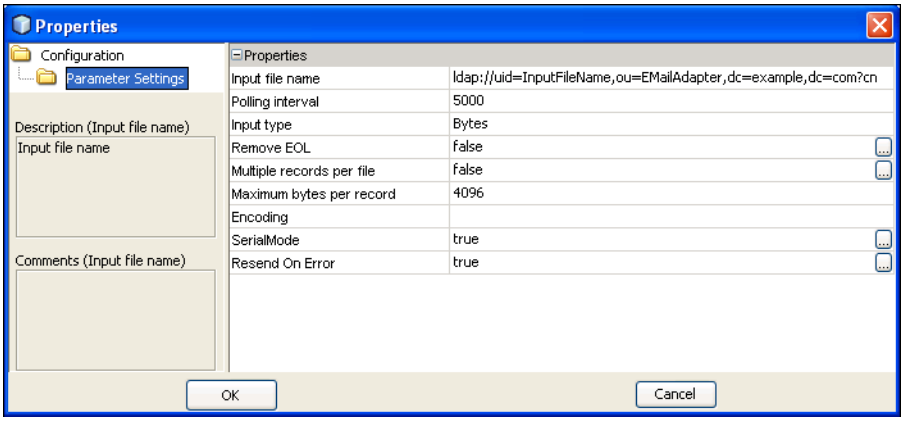
The correct path to the property value in the LDAP server depends on the directory structure. Do not include the backslash character (\) in an LDAP URL. RFC 2255 defines the format of LDAP URLs. You can view the RFC at `http://www.ietf.org/rfc.html`.

▼ To Specify an LDAP URL for a Property

1 **In the NetBeans IDE, access the Properties dialog box that includes the property.**

2 **Enter an LDAP URL that points to the corresponding attribute in the LDAP server.**

In the following screen capture, the Input File Name property is set to an LDAP URL.



3 **Go to the LDAP server and enter the actual value.**

4 **When you deploy the application file, ensure that the LDAP server is running. If the LDAP server is not running, then the deployment will not succeed.**

Index