HP Cloud Service Automation

Software Version 4.50

LDAP Configuration Tool

Contents

Overview **2**

Configuration **2**

Configuration Details 2

Database and LDAP Configuration Properties File 2

Communicating with the Oracle or MS SQL Database Using SSL 8

Usage **11**

Command Line Options 11

Example Usage **12**

Sample config.properties Contents **13**

Sample config.properties.ldap Contents **14**

Known Issues **16**

Document release date: July 2015

Software release date: July 2015

Overview

The LDAP Configuration Tool is a command line tool for HP Cloud Service Automation (HP CSA) that creates or updates the LDAP configuration of an organization.

LDAP (Lightweight Directory Access Protocol) used by HP CSA is used to:

- Authenticate a user's login to the Cloud Service Management Console or Marketplace Portal
- Authenticate a user's access to information
- Authorize a user's access to information

The LDAP Configuration Tool provides the same actions available in the Cloud Service Management Console: configure LDAP for authentication to log in to HP CSA and configure LDAP to access information in HP CSA. To completely configure access to HP CSA, using the Cloud Service Management Console, you must also configure access control for an organization to authorize a user's access to information. Refer to the Cloud Service Management Console online help for more information about configuring access control.

Configuration

The LDAP Configuration Tool is located in `<csa_home>\Tools\LdapTool\` where `<csa_home>` is the directory in which HP CSA is installed.

Configuration Details

Database and LDAP Configuration Properties File

Database and LDAP configuration properties files are required by the LDAP Configuration Tool when creating or updating the LDAP configuration of an organization. These configuration properties files must be located in the same folder as the `ldap-tool.jar` file (`<csa_home>\Tools\LdapTool\`). Sample configuration properties files can be generated using the LDAP Configuration Tool (see *Generating Sample Configuration Properties Files"* for more information).

- **Database configuration properties file** – Required information used to communicate with the HP CSA database. In the examples used in this document, this file is named `config.properties`, but you can use a different name. To specify the file in the command line, use the `-c` or `--dbconfig` option.

  If you use the sample database configuration properties file, you must provide or update the property values. See *"Database Configuration Properties File Parameters"* for more information about the contents of this file. See *"Sample config.properties Contents"* for examples of this file.

- **LDAP configuration properties file** – Required information used to specify the LDAP configuration to be created or updated. In the examples used in this document, this file is named `config.properties.ldap`, but you can use a different name. To specify the file in the command line, use the `-l` or `--ldapconfig` option.

  All required properties (Hostname, Port, User Email, Group Membership, Manager Identifier, Manager Identifier Value, User Name Attribute and User Search Filter) must be provided in this file. If you use the sample LDAP configuration properties file, you must uncomment and provide values for

the required properties. See *"LDAP Configuration Properties File Parameters"* for more information about the contents of this file. See *"Sample config.properties.ldap Contents"* for examples of this file.

Generating Sample Configuration Properties Files

The `ldap-tool.jar` produces sample configuration properties files by executing the following at the command prompt:

```
"<csa_jre>\bin\java" –jar ldap-tool.jar –g
```

where `<csa_jre>` is the directory in which the JRE that is used by HP CSA is installed.

> **Note**
> Additional command line options are required if SSL is enabled between the Oracle database and HP CSA. See
> *Communicating with the Oracle or MS SQL Database Using SSL"* for more information.

The following sample configuration properties files are generated:

- `config.properties.ldap`
- `config.properties.mssql`
- `config.properties.oracle`
- `config.properties.postgresql`

In the current directory, copy the sample database configuration properties file that corresponds to the type of database you are using to a file named `config.properties`. For example, if you are using an Oracle database, make a copy of the `config.properties.oracle` file and rename it to `config.properties`. Update the contents of `config.properties` as needed, as described in the table below. The other sample database configuration files can be deleted. For example, if you are using an Oracle database, delete the MS SQL and PostgreSQL sample configuration files.

In the current directory, make a copy of the sample LDAP configuration properties file as a backup file. Then, edit the `config.properties.ldap` file, as necessary (you must uncomment and provide values for the required properties). See *"LDAP Configuration Properties File Parameters"* for more information about the properties.

Database Configuration Properties File Parameters

This table lists the parameters found in the database configuration file.

Table 1. Database Configuration Properties File Parameters

Property Name	Description

jdbc.driverClassName	The JDBC driver class.
	Examples
	• Oracle
	`jdbc.driverClassName=oracle.jdbc.driver.OracleDriver`
	• MS SQL
	`jdbc.driverClassName=net.sourceforge.jtds.jdbc.Driver`
	• PostgreSQL
	`jdbc.driverClassName=org.postgresql.Driver`

jdbc.dialect	The classname that allows JDBC to generate optimized SQL for a particular database.
	Examples
	• Oracle
	`jdbc.dialect=org.hibernate.dialect.OracleDialect`
	• MS SQL
	`jdbc.dialect=org.hibernate.dialect.SQLServerDialect`
	• PostgreSQL
	`jdbc.dialect=org.hibernate.dialect.PostgreSQLDialect`

jdbc.databaseUrl	The JDBC URL. When specifying an IPv6 address, it must be enclosed in square brackets (see examples below).
	Examples
	• Oracle (SSL not enabled)
	`jdbc.databaseUrl=jdbc:oracle:thin:@//127.0.0.1:1521/XE`
	• Oracle (SSL not enabled, using an IPv6 address):
	`jdbc.databaseUrl=jdbc:oracle:thin:@//[f000:253c::9c10:b4b4]:1521/XE`
	• Oracle (SSL enabled, HP CSA does not check the database DN)
	`jdbc.databaseUrl=jdbc:oracle:thin:@(DESCRIPTION= (ADDRESS_LIST= (ADDRESS=(PROTOCOL = TCPS)(HOST = <host>)(PORT = 1521))) (CONNECT_DATA =(SERVICE_NAME = ORCL)))`
	where `<host>` is the name of the system on which the Oracle database server is installed.
	• Oracle (SSL enabled, HP CSA checks the database DN)
	`jdbc.databaseUrl=jdbc:oracle:thin:@(DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCPS)(HOST = <host>)(PORT = 1521))) (CONNECT_DATA = (SERVICE_NAME = ORCL))(SECURITY=(SSL_SERVER_CERT_DN= "CN=abc,OU=dbserver,O=xyz,L=Sunnyvale,ST=CA,C=US")))`
	where `<host>` is the name of the system on which the Oracle database server is installed and the values for `SSL_SERVER_CERT_DN` are for the DN of the Oracle database server.
	• MS SQL (SSL not enabled)
	`jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=request`
	• MS SQL (SSL not enabled, using an IPv6 address)
	`jdbc.databaseUrl=jdbc:jtds:sqlserver://[::1]:1433/example;ssl=request`
	• MS SQL (SSL enabled)
	`jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=authenticate`
	• MS SQL (FIPS 140-2 compliant)
	`jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=authenticate`

jdbc.Username	The user name of the database user you configured for HP CSA after installing the database.

jdbc.password	The password for the database user. The password should be encrypted (see "*Encrypt a Password*" in the ***HP CSA Configuration Guide*** for instructions on encrypting passwords). An encrypted password is preceded by ENC without any separating spaces and is enclosed in parentheses. If you have configured HP CSA to be FIPS 140-2 compliant, encrypt this password after you have configured HP CSA to be FIPS 140-2 compliant (that is, you should use the updated encryption tools to encrypt the password). **Example** `jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)`
idmConfig.Url	The system on which HP CSA is installed. Default: `https://127.0.0.1:8444`
securityTransportUserName	The user used to authenticate HP CSA Legacy 3.x REST API calls. Default: `csaTransportUser`
securityTransportPassword	The password for the user used to authenticate HP CSA Legacy 3.x REST API calls. The password should be encrypted (see "*Encrypt a Password*" in the ***HP CSA Configuration Guide*** for instructions on encrypting passwords). An encrypted password is preceded by ENC without any separating spaces and is enclosed in parentheses. If you have configured HP CSA to be FIPS 140-2 compliant, encrypt this password after you have configured HP CSA to be FIPS 140-2 compliant (that is, you should use the updated encryption tools to encrypt the password). **Example** `securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)`
securityIdmTransport UserName	The user used to authenticate HP CSA Consumption REST API calls. Default: `idmTransportUser`
securityIdmTransport Password	The password for the user used to authenticate HP CSA Consumption REST API calls. The password should be encrypted (see "*Encrypt a Password*" in the ***HP CSA Configuration Guide*** for instructions on encrypting passwords). An encrypted password is preceded by ENC without any separating spaces and is enclosed in parentheses. If you have configured HP CSA to be FIPS 140-2 compliant, encrypt this password after you have configured HP CSA to be FIPS 140-2 compliant (that is, you should use the updated encryption tools to encrypt the password). **Example** `securityIdmTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)`

LDAP Configuration Properties File Parameters

This table lists the parameters found in the LDAP configuration file.

Table 2. LDAP Configuration Properties File Parameters

Property	Name
csa.ldap.hostname	**Required**. The fully-qualified LDAP server domain name (server.domain.com) or IP address. **Example** `ldap.xyz.com`
csa.ldap.port	**Required**. The port used to connect to the LDAP server. 389 for ldap and 636 for ldaps.
csa.ldap.ssl	Connection Security. If the LDAP server is configured to require ldaps (LDAP over SSL), set this property to `true`. If the LDAP server does not require ldaps, set this property to `false`.
csa.ldap.basedn	Base distinguished name. The Base DN is the top level of the LDAP directory that is used as the basis of a search. **Example** `DC=cirrus,DC=com`
csa.ldap.userid	The fully distinguished name of any user with authentication rights to the LDAP server. If the LDAP server does not require a User ID or password for authentication, this value can be omitted. **Example** `CN=csaldap,CN=Users,DC=cirrus,DC=com`
csa.ldap.password	Password of the User ID. If the LDAP server does not require a User ID or password for authentication, this value can be omitted. The password should be encrypted (see "*Encrypt a Password*" in the ***HP CSA Configuration Guide*** for instructions on encrypting passwords). An encrypted password is preceded by ENC without any separating spaces and is enclosed in parentheses. If you have configured HP CSA to be FIPS 140-2 compliant, encrypt this password after you have configured HP CSA to be FIPS 140-2 compliant (that is, you should use the updated encryption tools to encrypt the password). **Example** `ENC(A0Ell2PmN6ajnhlInJAnEumDDvCBvQLV)`
csa.ldap.useremail	**Required**. Designates the email address of the user to which to send email notifications. Common LDAP attribute names for email include `mail` and `email`. If the value for this attribute in the user object in LDAP is empty or not valid, the user for whom the value is empty or not valid does not receive email notifications. **Example** `mail`
csa.ldap.groupmembership	**Required**. Identifies a user as belonging to the group. Common LDAP attribute names that convey group membership include `member` and `uniqueMember`. **Examples** • `member` • `member,uniqueMember`

csa.ldap.managerIdentifier	**Required**. Identifies the manager of the user. A common LDAP attribute name for a user's manager is `manager`. If the value for this attribute in the user object in LDAP is empty or not valid, approval policies that use the User Context Template will fail. **Example** **`manager`**
csa.ldap.managerIdentifierValue	**Required**. Describes the value of the manager identifier. A common value for the manager identifier in LDAP is the `dn` (distinguished name) of the manager's user object. If the manager's user object cannot be located based on the values for manager identifier and manager identifier value, approval policies that use the User Context Template will fail. **Example** `dn`
csa.ldap.userAvatar	LDAP attribute whose value is the URL to a user avatar image that will display for the logged in user in the Marketplace Portal. If no avatar is specified, a default avatar will be used. **Example** `avatar`
csa.ldap.userNameAttribute	**Required**. The name of the attribute of a user object that contains the username that will be used to log into the Cloud Service Management Console or Marketplace Portal. The value for this field can be determined by looking at one or more user objects in the LDAP directory to determine which attribute consistently contains a unique user name. **Example** `sAMAccountName`
csa.ldap.userSearchBase	LDAP container that contains the users. This value must be relative to the Base DN. **Example** `cn=Users`
csa.ldap.userSearchFilter	**Required**. Specifies the general form of the LDAP query used to identify users during login. It must include the pattern {0}, which represents the user name entered by the user when logging in to the Cloud Service Management Console or Marketplace Portal. The filter is generally of the form `<attribute>= {0}`, with `<attribute>` typically corresponding to the value entered for User Name Attribute. **Example** `sAMAccountName={0}`
csa.ldap.searchSubtree	When a user logs in to the Cloud Service Management Console or Marketplace Portal, the LDAP directory is queried to find the user's account. The Search Subtree setting controls the depth of the search under User Search Base. If you want to search for a matching user in the User Search Base and all subtrees under the User Search Base, set this property to `yes`. If you want to restrict the search for a matching user to only the User Search Base, excluding any subtrees, set this property to `no`. **Examples** • `yes` • `no`

Communicating with the Oracle or MS SQL Database Using SSL

If SSL is enabled between HP CSA and the Oracle or MS SQL database, additional command line options might be required and the URL property in the database properties file must be configured correctly.

Table 3. Oracle: HP CSA does not check the database DN and client authentication is enabled

Database Type	Oracle
Configuration Options	• SSL is enabled • HP CSA does NOT check the database DN • Client authentication is enabled

Command Line Option(s)	`-Djavax.net.ssl.keyStore="<certificate_key_file>"`
	`-Djavax.net.ssl.keyStorePassword=<certificate_key_file_password>`
	`-Djavax.net.ssl.keyStoreType=<certificate_key_file_type>`
	
	where:
	• `<certificate_key_file>` is the same keystore file defined by the certificate-keyfile attribute in the ssl element of the `<csa_home>\jboss\standalone\configuration\standalone.xml` file (for example, `<csa_home>\jboss\standalone\configuration\.keystore`),
	• `<certificate_key_file_password>` is the password to the keystore file (for example, changeit), and
	• `<certificate_key_file_type>` is the keystore type (for example, JKS or PKCS12)

jdbc.databaseURL Value	`jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST= (ADDRESS=(PROTOCOL = TCPS)(HOST = <host>)(PORT = 1521))) (CONNECT_DATA =(SERVICE_NAME = ORCL)))` where `<host>` is the name of the system on which the Oracle database server is installed

Table 4. Oracle: HP CSA does not check the database DN and client authentication is not enabled

Database Type	Oracle
Configuration Options	• SSL is enabled • HP CSA does NOT check the database DN • Client authentication is NOT enabled
Command Line Option(s)	<none>
jdbc.databaseURL Value	`jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST= (ADDRESS=(PROTOCOL = TCPS)(HOST = <host>)(PORT = 1521))) (CONNECT_DATA =(SERVICE_NAME = ORCL)))` where `<host>` is the name of the system on which the Oracle database server is installed

Table 5. Oracle: HP CSA checks the database DN and client authentication is enabled

Database Type	Oracle
Configuration Options	• SSL is enabled • HP CSA checks the database DN • Client authentication is enabled

Command Line Option(s)	`-Doracle.net.ssl_server_dn_match=true`
	`-Djavax.net.ssl.keyStore="<certificate_key_file>"`
	`-Djavax.net.ssl.keyStorePassword=<certificate_key_file_password>`
	`-Djavax.net.ssl.keyStoreType=<certificate_key_file_type>`
	
	where:
	• `<certificate_key_file>` is the same keystore file defined by the certificate-keyfile attribute in the ssl element of the `<csa_home>\jboss-as\standalone\configuration\standalone.xml` file (for example, `<csa_home>\jboss-as\standalone\configuration\.keystore`),
	• `<certificate_key_file_password>` is the password to the keystore file (for example, changeit), and
	• `<certificate_key_file_type>` is the keystore type (for example, JKS or PKCS12)

jdbc.databaseURL Value	`jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCPS)(HOST =<host>)(PORT = 1521))) (CONNECT_DATA = (SERVICE_NAME = ORCL))(SECURITY=(SSL_SERVER_CERT_DN= "CN=abc,OU=dbserver,O=xyz,L=Sunnyvale,ST=CA,C=US")))` where: • `<host>` is the name of the system on which the Oracle database server is installed and • the values for `SSL_SERVER_CERT_DN` are for the DN of the Oracle database server

Table 6. Oracle: HP CSA checks the database DN and client authentication is not enabled

Database Type	Oracle
Configuration Options	• SSL is enabled • HP CSA checks the database DN • Client authentication is NOT enabled
Command Line Option(s)	<none>
jdbc.databaseURL Value	`jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCPS)(HOST =<host>)(PORT = 1521))) (CONNECT_DATA = (SERVICE_NAME = ORCL))(SECURITY=(SSL_SERVER_CERT_DN= "CN=abc,OU=dbserver,O=xyz,L=Sunnyvale,ST=CA,C=US")))` where: • `<host>` is the name of the system on which the Oracle database server is installed and • the values for `SSL_SERVER_CERT_DN` are for the DN of the Oracle database server

Table 7. MS SQL

Database Type	MS SQL
Configuration Options	• SSL is enabled
Command Line Option(s)	<none>
jdbc.databaseURL Value	`jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=authenticate`

Usage

Command Line Options

The command options and sub-options for the LDAP Configuration Tool are shown in the following table.

Option	Option Description	Sub-options Associated with the Option	Sub-option Description
`-h, --help`	Display syntax and usage information.	none	
`-g, --generate`	Generate sample database and LDAP configuration properties files.	none	
`-o, --organization <organization Identifier>`	Name of the organization for which the LDAP configuration information needs to be created or modified	`-c, --dbconfig <config_filename>`	**Required**. The database configuration property filename. This file must be located in the same folder as the `ldap-tool.jar` file (`<csa_home>\Tools\LdapTool\`).
		`-l, --ldapconfig`	**Required**. The LDAP configuration property filename. This file must be located in the same folder as the `ldap-tool.jar` file (`<csa_home>\Tools\LdapTool\`).
		`-j, --jars <jar filenames>`	**Required** if you are using an Oracle database. Load Oracle JDBC JAR files. Note that jar filenames are separated by spaces. This file must be located in the same folder as the `ldap-tool.jar` file (`<csa_home>\Tools\LdapTool\`). If you are using an MS SQL or PostgreSQL database, you do not need to specify this option.

To list the supported options, invoke the LDAP Configuration Tool from the command line as follows:

```
java.exe -jar ldap-tool.jar –h

usage: java -jar ldap-tool.jar

       java -jar ldap-tool.jar -o [organization Identifier] [-c [database
      configuration properties file name] -j [oracle jar file location]] -l [ldap
      configuration properties file name]]

       java -jar ldap-tool.jar –h

       java -jar ldap-tool.jar –g
```

```
LDAP tool - The LDAP tool can be used to create or update LDAP configuration for an
organization.
Only a user with CSA administrator role will be able to run this tool.

-c,--dbconfig <config property file>    The database config property file name.
-g,--generate                           Generate sample input config properties file
                                        for database and ldap.
-h,--help                               Print this usage information.
-j,--jars <Oracle JARs>                 List of Oracle JDBC drivers, each separated by
                                        a space. This is required if the database is
                                        Oracle
-l,--ldapconfig <config property file>  The LDAP config property file name.
-o,--organizationIdentifier             Name of the organization for which LDAP
                                        configuration needs to be created or updated.
```

Example Usage

Note
When running the LDAP Configuration Tool to create or update the LDAP configuration for an organization, you are prompted for a username and password. This user MUST be assigned to the CSA Administrator role. Users who are not assigned to this role cannot create or update the LDAP configuration for an organization.

Note
Additional command line options are required if SSL is enabled between the Oracle database and HP CSA. See
Communicating with the Oracle or MS SQL Database Using SSL for more information.

Examples for Oracle (SSL is not Enabled)

- Display the LDAP Configuration Tool help:

  ```
  "<csa_jre>\bin\java" -jar ldap-tool.jar –h
  ```

- Generate sample configuration properties files:

  ```
  "<csa_jre>\bin\java" -jar ldap-tool.jar –g
  ```

- Create/update LDAP configuration for an organization:

  ```
  "<csa_jre>\bin\java" -jar ldap-tool.jar –o orgIdentifier–c config.properties -j
  ojdbc6.jar –l config.properties.ldap
  ```

Examples for MS SQL and PostgreSQL

- Display the LDAP Configuration Tool help:

  ```
  "<csa_jre>\bin\java" -jar ldap-tool.jar –h
  ```

- Generate sample configuration properties files:

```
"<csa_jre>\bin\java" -jar ldap-tool.jar –g
```

- Create/update LDAP configuration for an organization:

```
"<csa_jre>\bin\java" -jar ldap-tool.jar –o orgIdentifier –c config.properties  –l
config.properties.ldap
```

Sample config.properties Contents

Oracle (SSL not enabled)

```
jdbc.driverClassName=oracle.jdbc.driver.OracleDriver
jdbc.databaseUrl=jdbc:oracle:thin:@//127.0.0.1:1521/XE
jdbc.username=csa
jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)
jdbc.dialect=org.hibernate.dialect.OracleDialect
idmConfig.Url=https://127.0.0.1:8444
securityTransportUserName=csaTransportUser
securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)
securityIdmTransportUserName=idmTransportUser
securityIdmTransportPassword=ENC(lDdh98Kfe76op8lhjE0El897klRCB532lsb)
```

MS SQL (SSL not enabled)

```
jdbc.driverClassName=net.sourceforge.jtds.jdbc.Driver
jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=request
jdbc.username=csa
jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)
jdbc.dialect=org.hibernate.dialect.SQLServerDialect
idmConfig.Url=https://127.0.0.1:8444
securityTransportUserName=csaTransportUser
securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)
securityIdmTransportUserName=idmTransportUser
securityIdmTransportPassword=ENC(lDdh98Kfe76op8lhjE0El897klRCB532lsb)
```

MS SQL (SSL enabled)

```
jdbc.driverClassName=net.sourceforge.jtds.jdbc.Driver
jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=authenticate
jdbc.username=csa
jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)
jdbc.dialect=org.hibernate.dialect.SQLServerDialect
idmConfig.Url=https://127.0.0.1:8444
securityTransportUserName=csaTransportUser
securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)
```

```
securityIdmTransportUserName=idmTransportUser
securityIdmTransportPassword=ENC(lDdh98Kfe76op8lhjE0El897klRCB532lsb)
```

MS SQL (FIPS 140-2 compliant)

```
jdbc.driverClassName=net.sourceforge.jtds.jdbc.Driver
jdbc.databaseUrl=jdbc:jtds:sqlserver://127.0.0.1:1433/example;ssl=authenticate
jdbc.username=csa
jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)
jdbc.dialect=org.hibernate.dialect.SQLServerDialect
idmConfig.Url=https://127.0.0.1:8444
securityTransportUserName=csaTransportUser
securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)
securityIdmTransportUserName=idmTransportUser
securityIdmTransportPassword=ENC(lDdh98Kfe76op8lhjE0El897klRCB532lsb)
```

PostgreSQL

```
jdbc.driverClassName=org.postgresql.Driver
jdbc.databaseUrl=jdbc:postgresql://127.0.0.1:5432/csadb
jdbc.username=csadbuser
jdbc.password=ENC(fc5e38d38a5703285441e7fe7010b0)
jdbc.dialect=org.hibernate.dialect.PostgreSQLDialect
idmConfig.Url=https://127.0.0.1:8444
securityTransportUserName=csaTransportUser
securityTransportPassword=ENC(rlbE840uFDlert5441fe70jkY)
securityIdmTransportUserName=idmTransportUser
securityIdmTransportPassword=ENC(lDdh98Kfe76op8lhjE0El897klRCB532lsb)
```

Sample config.properties.ldap Contents

```
csa.ldap.hostname=172.16.200.50
csa.ldap.port=389
csa.ldap.ssl=false
csa.ldap.basedn=DC=cirrus,DC=com
csa.ldap.userid=CN=csaldap,CN=Users,DC=cirrus,DC=com
csa.ldap.password=ENC(A0Ell2PmN6ajnhlInJAnEumDDvCBvQLV)
csa.ldap.useremail=mail
csa.ldap.groupmembership=member
csa.ldap.managerIdentifier=manager
csa.ldap.managerIdentifierValue=dn
csa.ldap.userAvatar=avatar
csa.ldap.userNameAttribute=sAMAccountName
csa.ldap.userSearchBase=
csa.ldap.userSearchFilter=sAMAccountName={0}
csa.ldap.searchSubtree=no
```

Generated Sample LDAP Configuration Properties File

```
# Sample properties file for LDAP configuration in CSA.

# Required. The fully-qualified LDAP server domain name (server.domain.com) or IP
address.
# Example: ldap.xyz.com
#csa.ldap.hostname=

# Required. The port used to connect to the LDAP server. 389 for ldap and 636 for
ldaps.
# Example: 389
#csa.ldap.port=

# Required. This LDAP attribute designates the email address of the user to which to
send email notifications. Common LDAP attribute names for email include mail and
email.
# If the value for this attribute in the user object in LDAP is empty or not valid,
the user for whom the value is empty or not valid does not receive email
notifications.
# Example: mail
#csa.ldap.useremail=

# Required. This LDAP attribute identifies a user as belonging to the group. Common
LDAP attribute names that convey group membership include member and uniqueMember.
# Example: member,uniqueMember
#csa.ldap.groupmembership=

# Required. This LDAP attribute identifies the manager of the user. A common LDAP
attribute name for a user's manager is manager. If the value for this
# attribute in the user object in LDAP is empty or not valid, approval policies that
use the User Context Template will fail.
# Example: manager
#csa.ldap.managerIdentifier=

# Required. This LDAP attribute describes the value of the manager identifier.
# A common value for the manager identifier in LDAP is the dn (distinguished name) of
the manager's user object.
# If the manager's user object cannot be located based on the values for manager
identifier and manager identifier value, approval policies that use the User Context
Template will fail.
# Example: dn
#csa.ldap.managerIdentifierValue=

# Required. The name of the attribute of a user object that contains the username that
will be used to log into the Cloud Service Management Console or Marketplace Portal.
# The value for this field can be determined by looking at one or more user objects in
the LDAP directory to determine which attribute consistently contains a unique user
name.
# Example: sAMAccountName
#csa.ldap.userNameAttribute=
```

```
# Required. Specifies the general form of the LDAP query used to identify users during
login.
# It must include the pattern {0}, which represents the user name entered by the user
when logging in to the Cloud Service Management Console or Marketplace Portal. The
filter is generally of the form <attribute>= 0}, with <attribute> typically
corresponding to the value entered for User Name Attribute.
# Example: sAMAccountName={0}
#csa.ldap.userSearchFilter=

# Connection Security. If the LDAP server is configured to require ldaps (LDAP over
SSL), set this attribute to true.
# Example: false
#csa.ldap.ssl=

# Base distinguished name. The Base DN is the top level of the LDAP directory that is
used as the basis of a search.
# Example: DC=dom,DC=com
#csa.ldap.basedn=

# The fully distinguished name of any user with authentication rights to the LDAP
server. If the LDAP server does not require a User ID or password for authentication,
this value can be omitted.
# Example: CN=ldap,CN=Users,DC=dom,DC=com
#csa.ldap.userid=

# Password of the User ID. If the LDAP server does not require a User ID or password
for authentication, this value can be omitted.
# Example: password
#csa.ldap.password=

# LDAP attribute whose value is the URL to a user avatar image that will display for
the logged in user in the Marketplace Portal. If no avatar is specified, a default
avatar will be used.
# Example: avatar
#csa.ldap.userAvatar=

# The LDAP container that contains users. This value must be relative to the Base DN.
# Example:ou=People
#csa.ldap.userSearchBase=

# When a user logs in to the Cloud Service Management Console or Marketplace Portal,
the LDAP directory is queried to find the user's account.
# The Search Subtree setting controls the depth of the search under User Search Base.
# If you want to search for a matching user in the User Search Base and all subtrees
under the User Search Base, set the value of this attribute to yes.
# If you want to restrict the search for a matching user to only the User Search Base,
excluding any subtrees, set the value of this attribute to no.
# Example: yes
#csa.ldap.searchSubtree=
```

Known Issues

None.