Oracle Client 23ai LDAP URL Syntax

LDAP syntax for easy application configuration

August, 2024, Version 1.0

Public

Purpose statement

This document provides an overview of features and enhancements included in Oracle Database 23ai. It is intended solely to help you assess the business benefits of upgrading and planning for the implementation and upgrade of the product features described.

Disclaimer

This document in any form, software, or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement, nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described in this document remains at the sole discretion of Oracle. Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Table of contents

Introduction

Improved connection string syntax in Oracle Client 23ai makes it easier to use LDAP for Directory Name resolution when connecting to Oracle Database. The new feature extends Oracle Database's Easy Connect syntax to support LDAP and LDAPS connections. This removes the need for external LDAP configuration files, making LDAP easier for developers to use.

Directory Naming Overview

To connect to Oracle Database, applications use a connect descriptor containing information such as the database host and database service name. The connect descriptor can be supplied in various ways. For example, application code can pass a connect identifier that is mapped to a connect descriptor stored in a local *tnsnames.ora* configuration file. An alternative is for the connect descriptor to be stored in an external mapping service. One of the available external services is Directory Naming, which allows an application connect identifier to be mapped to a connect descriptor contained in an LDAP-compliant directory server such as Oracle Internet Directory (OID), Oracle Unified Directory (OUD), or Microsoft Active Directory.

Directory Naming centralizes network names and addresses in a single place, facilitating easy administration of name changes and updates. This eliminates the burden on administrators to manage connect descriptor updates to *tnsnames.ora* files stored on a large network of client machines.

LDAP URL Introduction

LDAP name lookup for Oracle Database is traditionally configured on application hosts with the help of external *ldap.ora* and *sqlnet.ora* files containing directory server names, ports and connection context.

Oracle Client 23ai introduces an additional way of LDAP lookup using a URL as the application connection string. The URL can contain values that previously needed to be stored in *ldap.ora* and *sqlnet.ora* files. The LDAP URL syntax can be used to connect to any database version from applications that are using Oracle Client 23ai libraries.

LDAP URL Syntax

The Oracle Client 23ai LDAP URL syntax is:

```
protocol://host[:port]/alias[,context][?parameter1=value1{&parameter2=value2...}]
```

For example:

```
ldaps://mydirserver.example.com/sales
```

The following table explains the options. The protocol, host name, and alias are mandatory.

Option	Description
protocol	The protocol can be either `ldap` or `ldaps`. The `ldaps` protocol uses TLS.
host	The hostname where the LDAP directory server is running.
port	Optional port number for the LDAP connection. The default port for the LDAP protocol is 389, and for LDAPS is 636.
alias	The LDAP entry to obtain the database connect descriptor from. This entry should be contained in the `OracleContext` container of a given Context.
context	Optional directory naming context containing `OracleContext`. For example, a context can be `cn=OracleContext,dc=example,dc=com`.

	The default value for this is `cn=OracleContext`.
parameters	Optional parameters are name-value pairs that define the LDAP connection. Parameters are described in the next table.

Optional URL Parameters

The LDAP URL syntax supports four optional, case-insensitive parameters that define the behaviour of connection. The delimiter “`?`” denotes the start of parameters. Individual parameters are separated by the delimiter “&”. The parameters are position independent. Whitespace between parameters is ignored.

Note that if a wallet is needed for TLS or mTLS database connections, the wallet location should be specified in a *sqlnet.ora* file or in the connect descriptor `WALLET_LOCATION` parameter stored in the LDAP server entry, not in the LDAP URL `WALLET_LOCATION` parameter.

Parameter Name	Description
`DIRECTORY_SERVER_TYPE`	The directory which will be used for LDAP based name look up. The value can be *OID* or *AD*. If you are using OUD, leave this value at its default, since the naming for OUD and OID is the same. The default is *OID*.
`AUTHENTICATE_BIND`	Specifies whether the LDAP naming adapter should attempt to authenticate using a specified wallet when connecting to the LDAP directory. If *TRUE*, then the LDAP connection is authenticated using a wallet whose location must be specified in the `WALLET_LOCATION` parameter. If *FALSE*, then the LDAP connection is established using an anonymous bind. The default value is *FALSE*.
`AUTHENTICATE_BIND_METHOD`	Specifies the authentication method that the client LDAP naming adapter should use while connecting to the LDAP directory to resolve a connect identifier. You can store the directory entry DN and password in an Oracle wallet. When the client connects to the LDAP server, it is authenticated using the credentials stored in this wallet. The wallet trust store must contain root certificates issued by the certificate authority of the LDAP server. The LDAP naming adapter uses the `oracle.ldap.client.dn` and `oracle.ldap.client.password` entries from the wallet for authenticating to the LDAP server. If these entries are not present, then the client attempts an anonymous authentication using TLS or LDAPS. For example: `AUTHENTICATE_BIND_METHOD=ldaps_simple_auth`
`WALLET_LOCATION`	Specifies the directory where a wallet is stored. This wallet is used for making TLS connections to the LDAP server. This parameter is not applicable to the database connection. If `WALLET_LOCATION` is not provided in the URL then *sqlnet.ora* is checked for the wallet location. If `WALLET_LOCATION` is not set in *sqlnet.ora* then the operating system certificate store is used.

Use Cases

This section shows examples of Oracle Client 23ai LDAP URL syntax. The examples search the directory entry `cn=orcl,cn=OracleContext,dc=example,dc=com`

Example 1

Example of basic usage. If a user wallet is not provided in the URL string, the client library checks for a user wallet in the `sqlnet.ora` file. If one is not found, then the default operating system default certificate store is used:

```
ldaps://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com
```

With Oracle's python-oracledb driver, this might be used as the connection string like:

```
cs = "ldaps://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com"
connection = oracledb.connect(user="scott", password=pw, dsn=cs)
```

Example 2

Example with OID simple authentication. The credential for the LDAP bind operation is taken from a provided wallet:

```
ldaps://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com?
WALLET_LOCATION=/app/wallet&AUTHENTICATE_BIND=true&AUTHENTICATE_BIND_METHOD=LDAPS_SIMPLE_AUTH
```

For mutual TLS (mTLS) LDAP authentication:

```
ldaps://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com?WALLET_LOCATION=/ap
p/wallet&AUTHENTICATE_BIND=true
```

Example 3

Example with Active Directory simple authentication. The credential for the LDAP bind operation is taken from a provided wallet:

```
ldaps://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com?DIRECTORY_SERVER_TY
PE=AD&WALLET_LOCATION=/app/wallet&AUTHENTICATE_BIND=true&AUTHENTICATE_BIND_METHOD=LDAPS_SIMPL
E_AUTH
```

Active Directory with Windows native authentication (using Windows login credentials):

```
ldap://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com?DIRECTORY_SERVER_TYP
E=AD&AUTHENTICATE_BIND=true
```

Example 4

Example connection to an LDAP server with a clear text port:

```
ldap://ldapserver.example.com/cn=orcl,cn=OracleContext,dc=example,dc=com
```

Conclusion

Applications using Oracle Client 23ai libraries can take advantage of new, optional, simple URL syntax for LDAP server Directory Name resolution. This supports connection to any Oracle Database version. The new syntax makes using LDAP easier, removing the overhead of distributing traditional LDAP configuration files to client machines.

You may also be interested in the new Oracle Database Centralized Configuration Provider feature that allows connection and application configuration information to be stored in Azure App Configuration Store or Oracle Cloud Infrastructure (OCI) Object Storage.

References

Documentation: Specify LDAP Parameters Directly in a Connect Identifier

Technical brief: Configuring Oracle Database Clients for OID and OUD Directory Naming

Technical brief: Configuring Oracle Database Clients for Microsoft Active Directory Naming

Connect with us

Call **+1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com facebook.com/oracle twitter.com/oracle