

Practical Exercise: Configure LDAP as Secondary User Store

Training Objective

Learn how to add a secondary user store to the platform so that the developers can use their existing credentials to log in and use WSO2 API Manager.

Business Scenario

PizzaShack wants to accelerate its digital transformation by providing an API Management platform to its developers.

High-Level Steps

- Configuring a Read/Write LDAP as a Secondary User Store.

Detailed Instructions

[WSO2 Identity Server](#) uses an embedded Read/Write LDAP as the primary user store and that will be added as the secondary user store for the API Manager as an example here.

Follow the steps below to configure the LDAP as a Secondary User Store.

1. Offset WSO2 Identity Server (Version 5.11.0) by 1.
2. Start WSO2 API Manager and WSO2 IS servers.
3. Log in to the Management Console: <https://localhost:9443/carbon> as admin user and click **Add** under the **User Stores** submenu in the **Main** menu.
4. The **Add New User Store** page opens.
5. In the User Store Manager Class list, select the type of user store you are creating.
 - a. Choose the **UniqueIDReadWriteLDAPUserStoreManager** for LDAP user stores to do both read and write operations.
6. Enter a unique domain name with no underscore (_) characters, and optionally enter a description for this user store:
 - a. Type: **wso2.com**
7. Enter values for the properties. (An example is given below. Replace them with values connected to your LDAP if you already have an LDAP). You can find the values related to **WSO2 IS Embedded LDAP** at **<IS_HOME>/repository/conf/user-mgt.xml** file.

| Property | Value |
|----------------|------------------------|
| Connection URL | ldap://localhost:10390 |

| | |
|-------------------------|-------------------------|
| Connection Name | uid=admin,ou=system |
| Connection Password | admin |
| User Search Base | ou=Users,dc=wso2,dc=org |
| User Entry Object Class | identityPerson |

8. Example

Add New User Store

User Store Manager

User Store Manager Class: `org.wso2.carbon.user.core.ldap.UniqueIDReadWriteLDAPUserStoreManager`
Depending on the class, properties needs to be defined.

Domain Name*: `wso2.com`

Description: `WSO2 IS Embedded LDAP`

Define Properties For

| Property Name | Property Value | Description |
|---------------------------|---|--|
| Connection URL * | <code>ldap://localhost:10390</code> | ② Connection URL |
| Connection Name * | <code>uid=admin,ou=system</code> | ② This should be in LDAP |
| Connection Password * | | ② Password of the connection |
| User Search Base * | <code>ou=Users,dc=wso2,dc=org</code> | ② DN of the connection |
| User Entry Object Class * | <code>identityPerson</code> | ② Object Class used for the connection |
| Username Attribute * | <code>uid</code> | ② Attribute used for the connection |
| User Search Filter * | <code>(&(objectClass=person)(uid=?))</code> | ② Filtering criteria |
| User List Filter * | <code>(objectClass=person)</code> | ② Filtering criteria |
| User ID Attribute * | <code>scimId</code> | ② Attribute used for the connection |
| User ID Search Filter * | <code>(&(objectClass=person)(uid=?))</code> | ② Filtering criteria |

Optional

Advanced

Update Cancel

- Validate your configuration by listing users. Click List under **User and Roles** and then select **Users**. You should see a list of employees pre-defined by WSO2 Instructors.
- Add a new user in your secondary user store and assign all Internal roles to it.

Add New User

Step 1: Enter Username and Password

Enter username

| | |
|-------------------|--|
| Domain | wso2.com ▼ |
| Username* | <input type="text" value="newuser"/> |
| Password* | <input type="password" value="*****"/> |
| Confirm Password* | <input type="password" value="*****"/> |

11. Log in to both Publisher and DevPortal using the newly created user.

For more information, refer to the [documentation](#).

Expected Outcome

By the end of this exercise, you should be able to add users to a Secondary User Store configured and log in to WSO2 API Manager using one of the LDAP credentials.