

Single Sign-on Configuration Guide -SAML
Oracle Banking APIs
Release 22.2.0.0.0

Part No. F72988-01

November 2022

ORACLE®

Single Sign-on Configuration Guide -SAML

November 2022

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway

Goregaon (East)

Mumbai, Maharashtra 400 063

India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001

www.oracle.com/financialservices/

Copyright © 2006, 2022, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Table of Contents

1. Preface	1-1
1.1 Intended Audience.....	1-1
1.2 Documentation Accessibility	1-1
1.3 Access to Oracle Support.....	1-1
1.4 Structure	1-1
1.5 Related Information Sources.....	1-1
2. Introduction	2-1
3. Configuration	3-1
3.1 Identity Provider Configuration at IDCS.....	3-1
3.2 SAML Authentication Provider configuration.....	3-6
3.3 SQL Authentication Provider configuration.....	3-9
3.4 OHS Configuration.....	3-13
3.5 Database Configuration	3-15
3.6 IDCS OAuth Integration	3-16
3.7 WebLogic configuration for OAuth.....	3-22
3.8 OBAPI configuration for OAuth.....	3-26
3.9 Default Admin Configuration.....	3-27
3.10 Logout Configurations.....	3-28

1. Preface

1.1 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

1.2 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

1.3 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

1.4 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describes following details:

- Introduction
- Preferences & Database
- Configuration / Installation.

1.5 Related Information Sources

For more information on Oracle Banking APIs Release 22.2.0.0.0, refer to the following documents:

- Oracle Banking APIs Installation Manuals

2. Introduction

This document covers step-by-step details on configuration required at IDCS side (Application and User) and WebLogic console configurations for SAML and SQL Authentication Providers. Document also includes the configuration required on OHS to enable different URL's for internal and external user login.

[Home](#)

3. Configuration

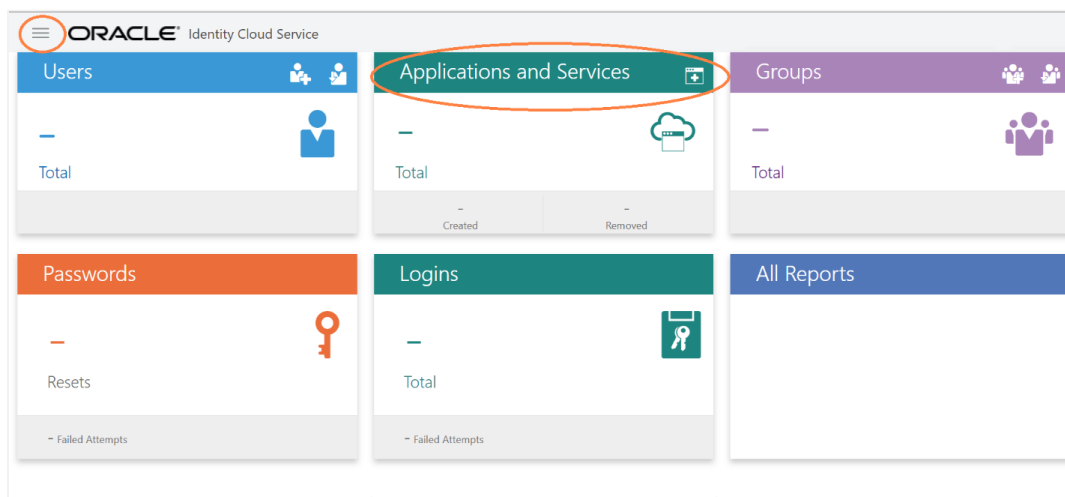
To enable SAML authentication it involves configuration at WebLogic server (console) and IDCS console.

3.1 Identity Provider Configuration at IDCS

Steps to configure Identity Provide at IDCS

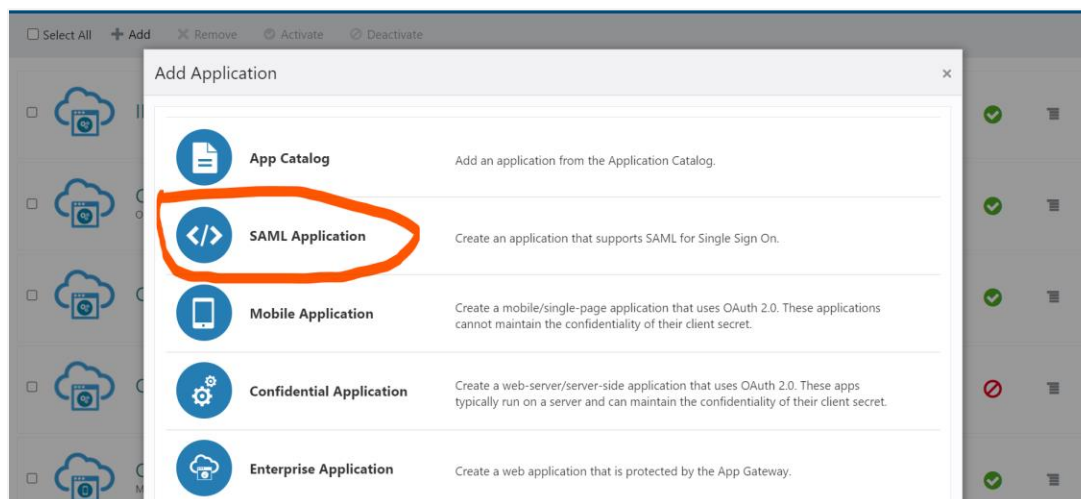
1. Login to Oracle Identity Cloud Service (IDCS) console with admin login. In dashboard click on **Add Application** in Application and Services widget or navigate through the breadcrumb menu as highlighted in screenshot.

Dashboard



2. In popup window select **SAML Application**.

Add Application



3. In **Add SAML Application** page provide below mentioned fields and click on **Next**.

- i. Name
- ii. Description

Add SAML Application

4. Fill below mentioned fields as per section.

- i. General
 - a. Entity Id: - A unique identifier / name for the service provider.
 - b. Assertion Consumer URL: - End point to which assertion will be sent by IDCS. Recommended URL format [<OHS_URL>/saml2/sp/acs/post](#)
e.g. [<PROTOCOL>://<OHS_HOST>:<OHS_PORT>/saml2/sp/acs/post](#)
[http://whf000xxx.bank.com:9999/saml2/sp/acs/post](#)
 - c. NameID Format: - Select value as “Unspecified”.
 - d. NameID Value:- Select value as “User Name”.

Add SAML Application

- ii. Advance Settings
 - a. Signed SSO :- Select value as “Assertion”
 - b. Enable Single Logout: - This field should be checked.
 - c. Logout Binding: - Select value as “Redirect”.
 - d. Single Logout URL: - End point which IDCS will make call to do single logout functionality.
Recommended URL format <OHS_URL>/digx-infra/sso-logout
[e.g. <PROTOCOL>://<OHS_HOST>:<OHS_PORT>/digx-infra/sso-logout](http://whf000xxx.bank.com:9999/digx-infra/sso-logout)
<http://whf000xxx.bank.com:9999/digx-infra/sso-logout>
 - e. Logout Response URL: -
Recommended URL format <OHS_URL>/digx-infra/sso-logout
[e.g. <PROTOCOL>://<OHS_HOST>:<OHS_PORT>/digx-infra/sso-logout](http://whf000xxx.bank.com:9999/digx-infra/sso-logout)
<http://whf000xxx.bank.com:9999/digx-infra/sso-logout>

Add SAML Application

Advanced Settings

This section contains additional configuration options.

Signed SSO

Include Signing Certificate in Signature

Signature Hashing Algorithm

Enable Single Logout

* Logout Binding

* Single Logout URL

* Logout Response URL

Encrypt Assertion

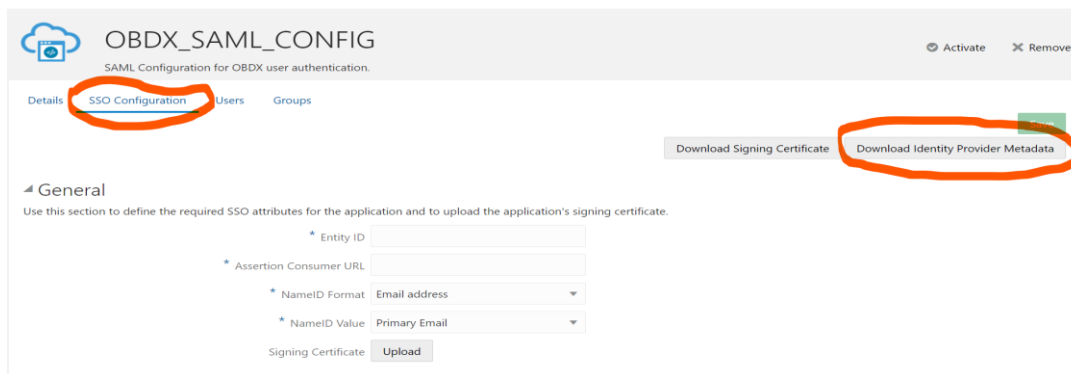
5. Click on **Finish / Save**.
6. Click on **Activate** button to activate your application.

Edit Application

The screenshot shows the 'Edit Application' page for 'OBDX_SAML_CONFIG'. At the top right, there are 'Activate' and 'Remove' buttons, with 'Activate' circled in orange. Below the application name, there are tabs for 'Details', 'SSO Configuration', 'Users', and 'Groups'. The 'Details' tab is active, showing 'App Details' with fields for 'Application Type' (SAML Application), 'Name' (OBDX_SAML_CONFIG), 'Description' (SAML Configuration for OBDX user authentication), and 'Application Icon'. A 'Save' button is circled in orange at the bottom right of the details section.

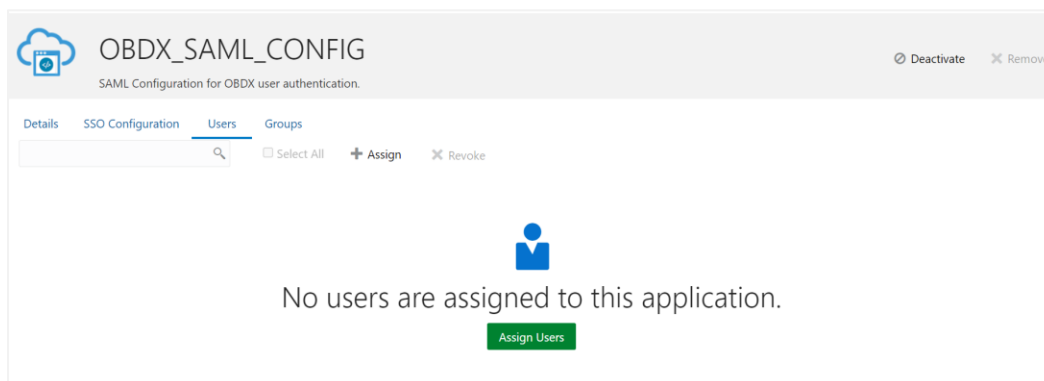
7. Navigate to Dashboard and search the application you have created.
8. Navigate to **SSO Configuration** tab and click on **“Download Identity Provider Metadata”**. Keep the downloaded xml file, it will be required to upload in WebLogic console. Same is explain in WebLogic console configuration steps.

Edit Application



9. Copy / FTP the downloaded IDC metadata xml file to WebLogic server using winscp / putty.
10. Navigate to **Users** tab in application to add the users related to application.
11. Click on **Assign Users** or **Assign (+)** button to search and add the users into application. If user is not available follow steps mentioned in Section 1.3 to create new user.

Edit Application



Assign Users

Assign Users ✕

i Please select up to 40 users to assign.

Select All 🔍

Selected: 1 [Clear Selection](#)

	First Name	Last Name	Email
<input type="checkbox"/>	Super	Admin	[REDACTED]
<input checked="" type="checkbox"/>	superadmin	superadmin	[REDACTED]

Page 1 of 1 (1-2 of 2 items) ⏪ < 1 > ⏩ OK

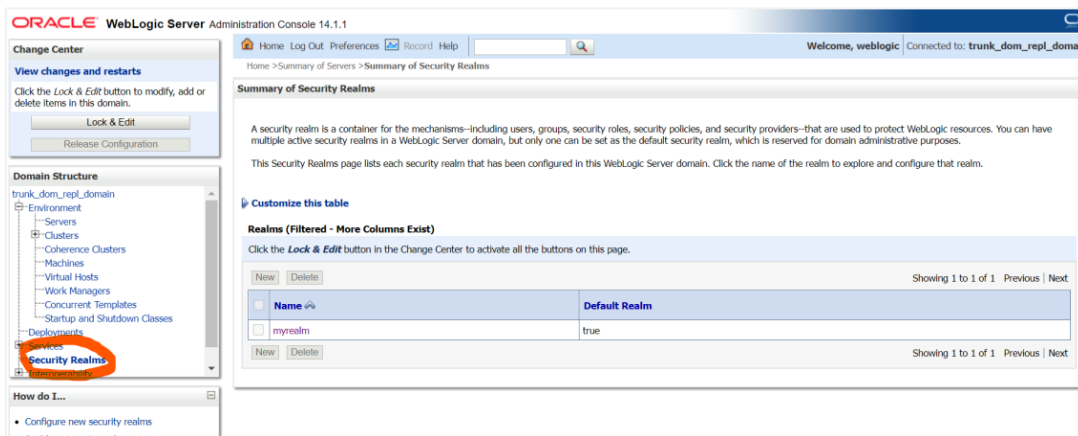
12. Logout from IDSC console.

3.2 SAML Authentication Provider configuration.

Steps to configure SAML Authentication Providers changes into WebLogic console.

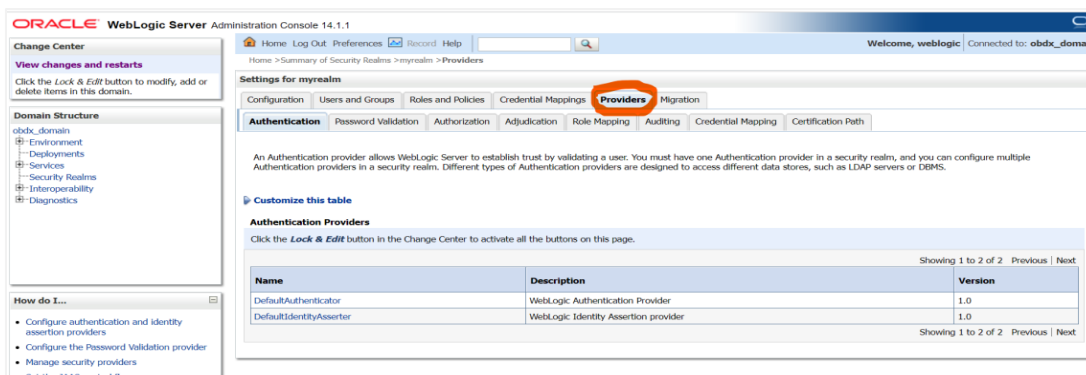
1. Login to WebLogic console with admin login and navigate to “Security Realms”.

Security Realms



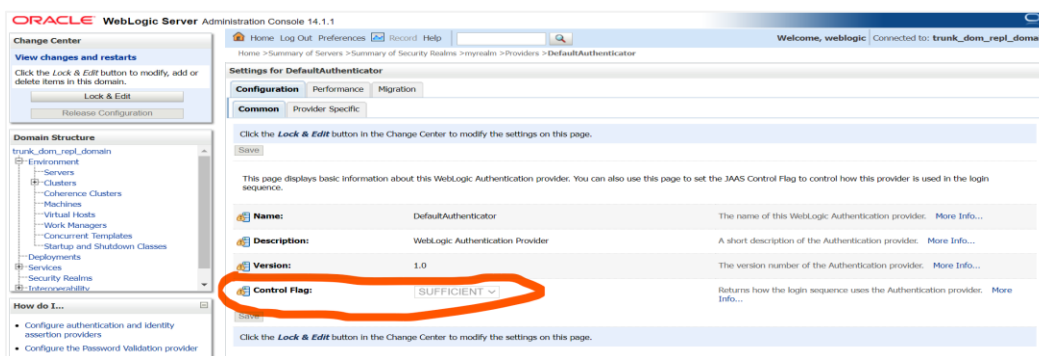
2. → Click on **myrealm** or your realm name present in screen. Navigate to “Providers” tab.

Providers



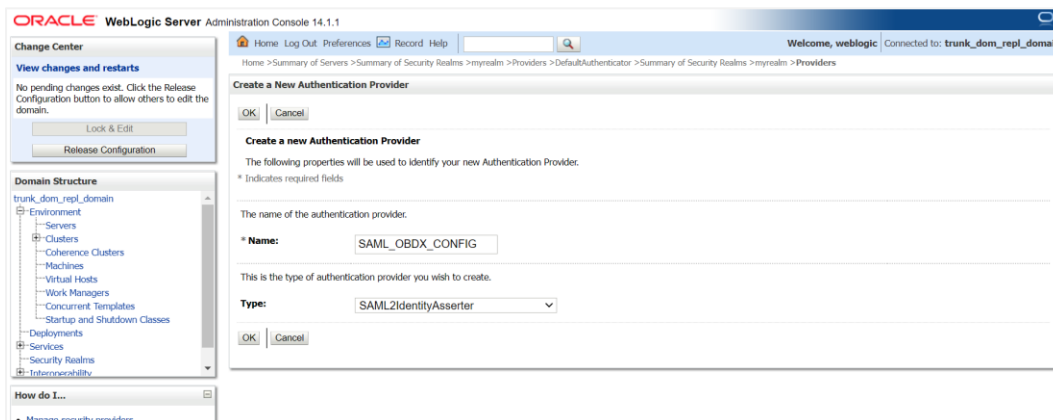
3. Select “DefaultAuthenticator” and change the Control Flag value to “SUFFICIENT”.

Default Authenticator



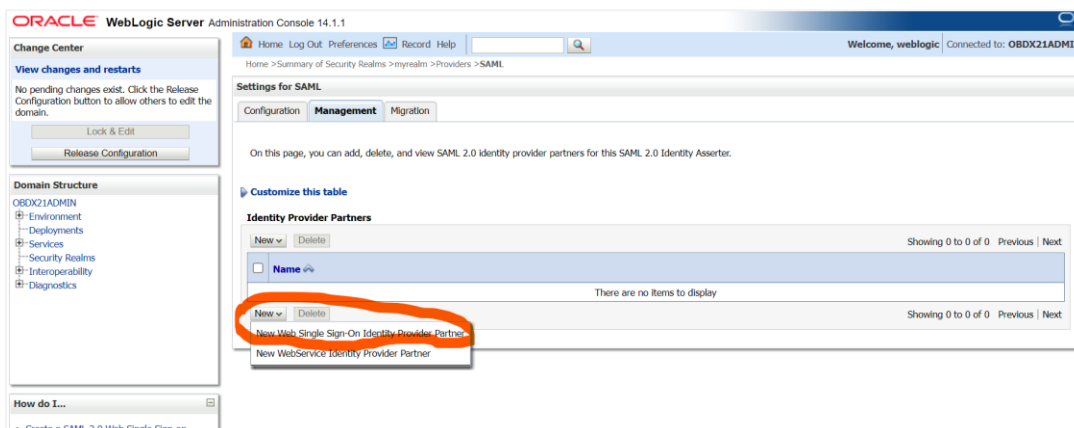
4. Again, navigate to “Security Realms” → myrealms → Providers and click on **New** button to create new Authentication Provider. Fill the below mentioned fields with appropriate values and click on **OK**.
 - i. Name: - Name of authentication provider.
 - ii. Type :- Select value as “SAML2IdentityAsserter”.

Create Authentication Provider



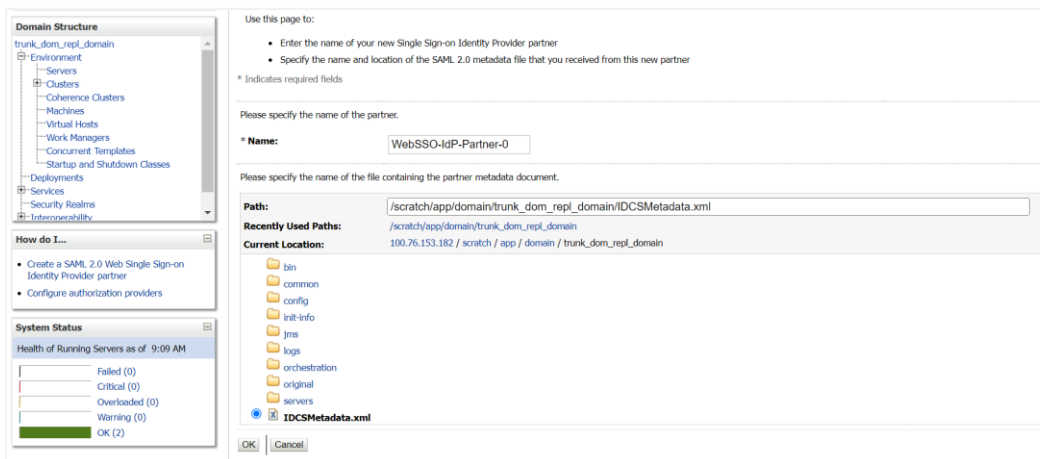
5. Restart Admin Server.
6. Login to WebLogic console and navigate to “Security Realms” → myrealms → Providers newly created authentication provider (e.g. SAML_OBAPI_CONFIG) and navigate to “Management” tab.
7. Click on **New** button to add the Identity Provider Partner and select “New Web Single Sign-On Identity Provider Partner”

Management



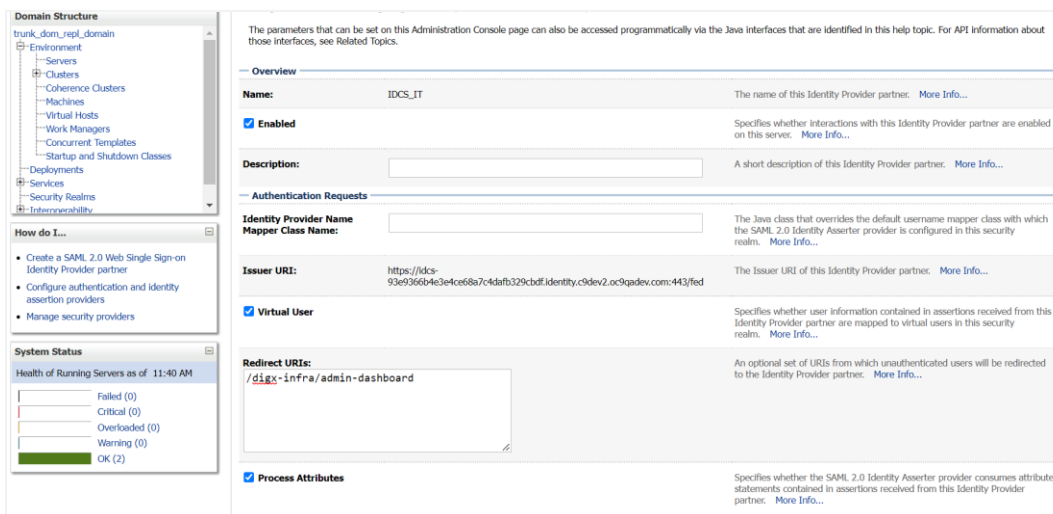
8. Provide the name for the identity partner and select the IDC metadata xml copied to WebLogic server. Click **OK** button to save.

Create a SAML 2.0 Web Single Sign-on Identity Provider Partner



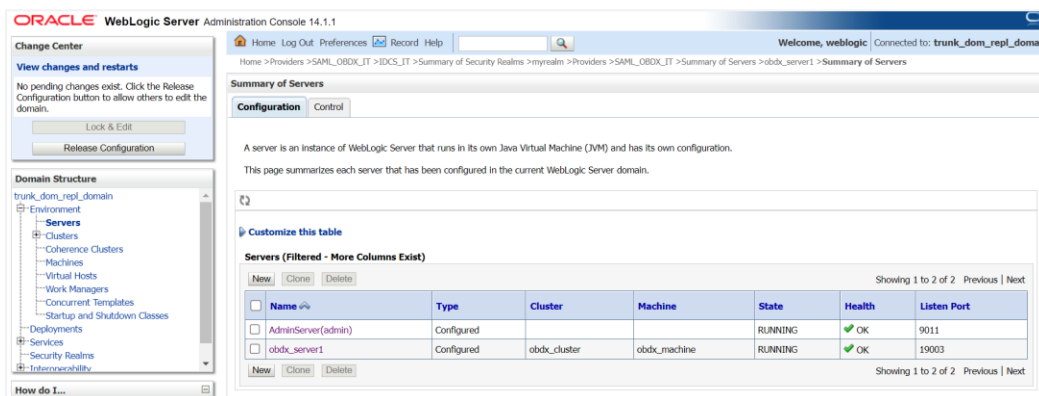
9. Open the newly added Identity Provider Partner and select below mentioned checkboxes and field and click on **Save**.
 - i. Enable: - Checked
 - ii. Virtual User: - Checked
 - iii. Redirect URIs: - /digx-infra/admin-dashboard

Settings for Create a SAML 2.0 Web Single Sign-on Identity Provider Partner



10. Navigate to “Environment” → “Servers” and select the server on which SSO authentication application will be deployed.

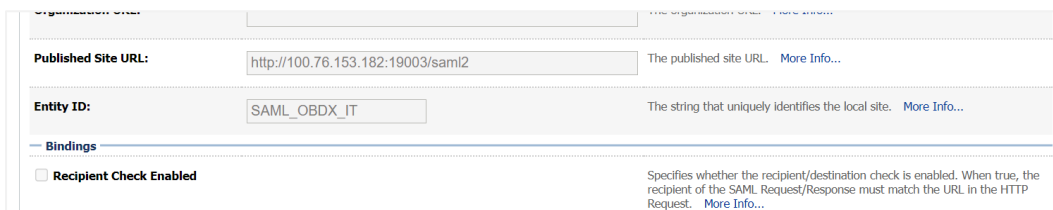
Servers



11. Navigate to “Federation Services” → “SAML 2.0 General” and provide values to below mentioned fields. Click on **Save**.

- i. Published Site URL: - Recommended URL format [<OHS URL>/saml2](#)
e.g. [<PROTOCOL>://<OHS_HOST>:<OHS_PORT>/saml2](#)
<http://whf000xxx.bank.com:9999/saml2>
- ii. Entity Id: - Value should match with [Entity Id](#) provided in SAML configuration in IDCS console.
- iii. Recipient Check Enabled: - unchecked.

SAML 2.0 General



12. Navigate to “Federation Services” → “SAML 2.0 Service Provider” and provide values to below mentioned fields and click on **Save**.

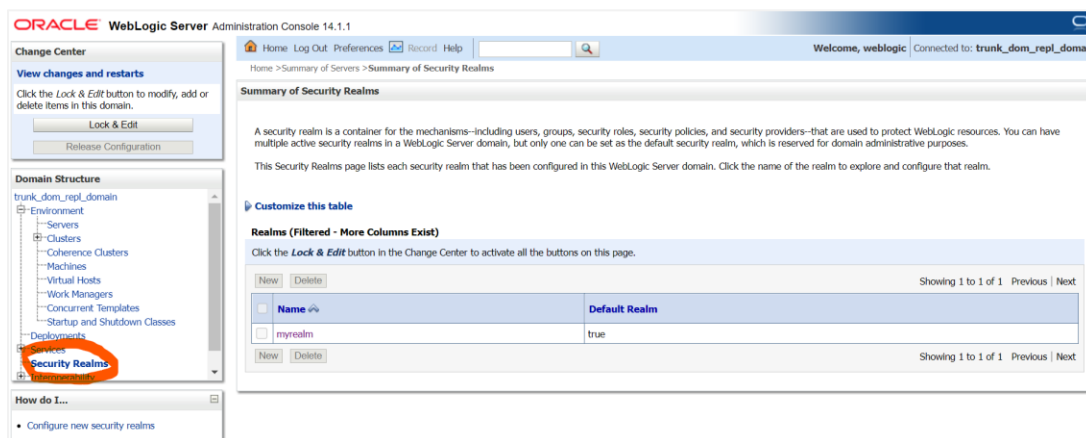
- i. Enabled: - Check box should be checked.
- ii. Preferred Binding: - Post
- iii. Default URL: - [<OHS_URL>/digx-infra/admin-dashboard](#)

3.3 SQL Authentication Provider configuration.

Steps to configure SQL Authentication Providers changes into WebLogic console.

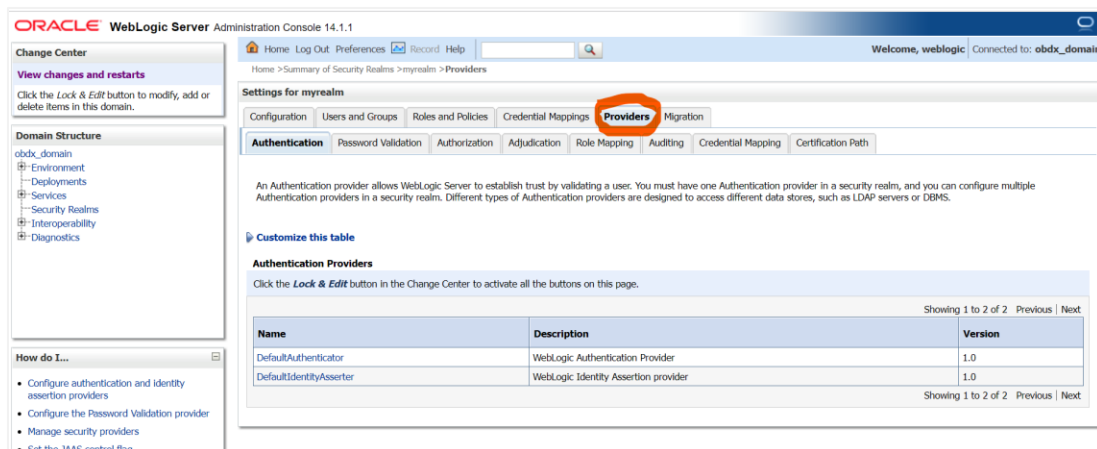
1. Login to WebLogic console with admin login and navigate to “**Security Realms**”.

Security Realms



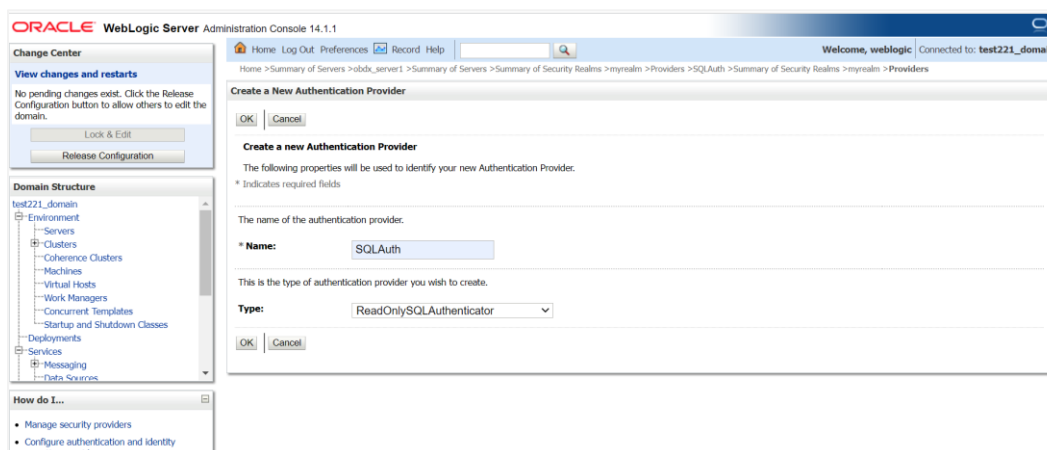
2. → Click on **myrealm** or your realm name present in screen. Navigate to “**Providers**” tab.

Providers



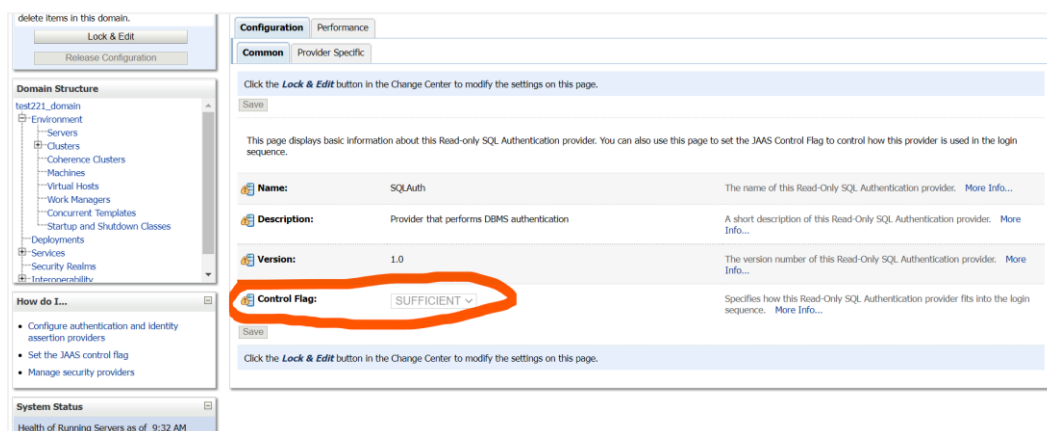
3. Click on **New** button to create new Authentication Provider. Fill the below mentioned fields with appropriate values and click on **OK**.
 - i. Name: - Name of authentication provider.
 - ii. Type :- Select value as “ReadOnlySQLAuthenticator”.

Create New Authentication Provider



4. Open newly created authentication provider (e.g. SQLAuth). Select the value of **Control Flag** as **“SUFFICIENT”**

Settings for Read Only SQL Authentication Provider



5. Navigate to **“Provider Specific”** tab to configuration related to SQL Authentication.
6. Provide the values to fields mentioned below with given value in case it is not auto populated.
 - i. Data Source Name: - NONXA
 - ii. SQL Get Users Password: - SELECT U_PASSWORD FROM USERS WHERE U_NAME = ?
 - iii. SQL User Exists: - SELECT U_NAME FROM USERS WHERE U_NAME = ?
 - iv. SQL List Users: - SELECT U_NAME FROM USERS WHERE U_NAME LIKE ?
 - v. SQL List Groups: - SELECT G_NAME FROM GROUPS WHERE G_NAME LIKE ?
 - vi. VI. SQL Group Exists: - SELECT G_NAME FROM GROUPS WHERE G_NAME = ?
 - vii. SQL Is Member: - SELECT G_MEMBER FROM GROUPMEMBERS WHERE G_NAME = ? AND G_MEMBER = ?

- viii. SQL List Member Groups: - SELECT G_NAME FROM GROUPMEMBERS WHERE G_MEMBER = ?
- ix. SQL Get User Description: - SELECT U_DESCRIPTION FROM USERS WHERE U_NAME = ?
- x. SQL Get Group Description: - SELECT G_DESCRIPTION FROM GROUPS WHERE G_NAME = ?

Settings for Read Only SQL Authentication Provider

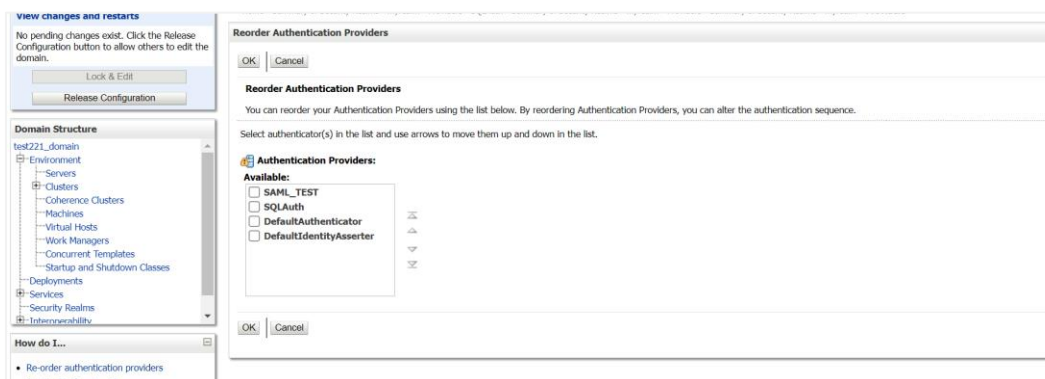
7. Click on **Save**.
8. Navigate to “Security Realms” → myrealms → Providers and click on **Reorder** button.

Authentication

9. Reorder the authentication providers as given below.

- i. SAML Authentication Provider
- ii. SQL Authentication Provider
- iii. Default Authenticator

Reorder Authentication Providers



10. Restart all the servers in domain including Admin Server.

****Note:** Accessing /saml2 uri from OHS (<OHS_URL>/saml2), /saml2 uri has to be proxy bypassed from OHS

3.4 OHS Configuration

Provides details on configuration required on OHS to enable different URL's for internal and external users. i.e authentication with OBAPI or external service provider.

1. Open obapi.conf file from OHS server. You can find the location of obapi.conf file from httpd.conf file.
2. Verify if proxypass URLs are configured in obapi.conf file. If not then add entries as mentioned in below format.

```
ProxyPassMatch "/digx(.*)" "<PROTOCOL>://<WL_HOST_NAME>:<WL_PORT>/digx$1"
ProxyPassReverse "/digx(.*)" "<PROTOCOL>://<WL_HOST_NAME>:<WL_PORT>/digx$1"
ProxyPassMatch "/saml2(.*)" "<PROTOCOL>://<WL_HOST_NAME>:<WL_PORT>/saml2$1"
ProxyPassReverse "/saml2(.*)" "<PROTOCOL>://<WL_HOST_NAME>:<WL_PORT>/saml2$1"
ProxyPassMatch "/digx(.*)" "http:// whf000xxx.bank.com:19003/digx$1"
ProxyPassReverse "/digx(.*)" "http:// whf000xxx.bank.com:19003/digx$1"
ProxyPassMatch "/saml2(.*)" "http:// whf000xxx.bank.com:19001/saml2$1"
ProxyPassReverse "/saml2(.*)" "http:// whf000xxx.bank.com:19001/saml2$1"
```

3. Add below virtual configuration into obapi.conf file.

```
##Virtual Hosts

Listen <PORT_1>

<VirtualHost *:<PORT_1>>

    ServerName <HOST_NAME>

    RewriteEngine On

    RewriteOptions inherit

    <Directory "${DocumentRoot}">

        Options FollowSymLinks

        AllowOverride all

    </Directory>

</VirtualHost>

Listen <PORT_2>

<VirtualHost *:<PORT_2>>

    ServerName <HOST_NAME>

    RewriteEngine On

    RewriteRule  "^(.*)/config\.js$"
"<SERVER_PROTOCOL>://<HOST_NAME>:<PORT_2>/framework/js/configurations/
config-admin.js" [R]

    <Directory "${DocumentRoot}">

        Options FollowSymLinks

        AllowOverride all

    </Directory>

</VirtualHost>
```

****Note:** Replace the <PORT_1> & <PORT_2> with the ports which are expose to outside world. Replace <SERVER_PROTOCOL> and <HOST_NAME> with appropriate values. E.g. http and whfxxx.sample.com (if hostname is not available then <HOST_NAME> value can be IP address.)

```
# All other request passed through this rules.
ProxyPassMatch "/digx(.*)" "http://whf00qiw.in.oracle.com:19001/digx$1"
ProxyPassReverse "/digx(.*)" "http://whf00qiw.in.oracle.com:19001/digx$1"
ProxyPassMatch "/saml2(.*)" "http://whf00qiw.in.oracle.com:19001/saml2$1"
ProxyPassReverse "/saml2(.*)" "http://whf00qiw.in.oracle.com:19001/saml2$1"

##Virtual Hosts
Listen 8888
<VirtualHost *:8888>
    ServerName whf00qiw.in.oracle.com
    RewriteEngine On
    RewriteOptions inherit

    <Directory "${DocumentRoot}">
        Options FollowSymLinks
        AllowOverride all
        #Require all granted
    </Directory>
</VirtualHost>

Listen 9999
<VirtualHost *:9999>
    ServerName whf00qiw.in.oracle.com
    RewriteEngine On
    RewriteRule "^(.*)/config\.js$" "http://whf00qiw.in.oracle.com:9999/framework/js/configurations/config-admin.js" [R]

    <Directory "${DocumentRoot}">
        Options FollowSymLinks
        AllowOverride all
        #Require all granted
    </Directory>
</VirtualHost>
```

4. Save obapi.conf file and restart ohs server.

3.5 Database Configuration

To enable SSO for external users below configuration need to be done in database.

1. To enable SSO authentication for user type / enterprise role execute below query on intended database environment. Replace <USER_TYPE> with the user type / enterprise role for which SSO authentication to be enabled.

```
UPDATE AUTH_CONFIG SET PROP_VALUE = 'External' WHERE PROP_ID = '<USER_TYPE>'
AND CATEGORY_ID = 'UserTypeConfig';
```

For example: - UPDATE AUTH_CONFIG SET PROP_VALUE = 'External' WHERE PROP_ID = 'administrator' AND CATEGORY_ID = 'UserTypeConfig';

- Execute below query for redirection after authentication from SSO service provider back to OBAPI. Replace the value of <OHS_URL_FOR_ADMIN_USER_LOGIN> with the OHS_URL with port enable for external / admin user login, the virtual host enabled in section 3.4, step 3.

```
INSERT INTO DIGX_FW_CONFIG_ALL_B (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER, EDITABLE, CATEGORY_DESCRIPTION) values
('SSO_PUBLIC_URL', 'dayoneconfig', '<OHS_URL_FOR_ADMIN_USER_LOGIN>', 'N', null,
'Public SSO URL', 'ofssuser', to_timestamp('29-09-22 10:05:56.000000000 AM', 'DD-MM-RR
fmHH12:fmMI:SSXFF AM'), 'ofssuser', to_timestamp('29-09-22 10:05:56.000000000 AM', 'DD-
MM-RR fmHH12:fmMI:SSXFF AM'), 'A', 1, 'N', null);
```

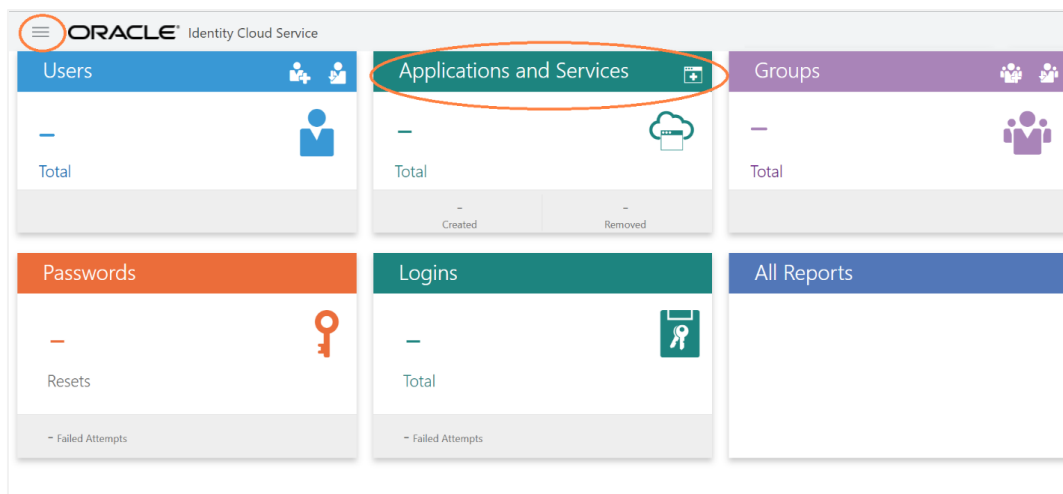
For Example: - INSERT INTO DIGX_FW_CONFIG_ALL_B (PROP_ID, CATEGORY_ID, PROP_VALUE, FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS, OBJECT_VERSION_NUMBER, EDITABLE, CATEGORY_DESCRIPTION) values ('SSO_PUBLIC_URL', 'dayoneconfig', 'http:// whf000xxx.bank.com:9999', 'N', null, 'Public SSO URL', 'ofssuser', to_timestamp('29-09-22 10:05:56.000000000 AM', 'DD-MM-RR fmHH12:fmMI:SSXFF AM'), 'ofssuser', to_timestamp('29-09-22 10:05:56.000000000 AM', 'DD-MM-RR fmHH12:fmMI:SSXFF AM'), 'A', 1, 'N', null);

3.6 IDCS OAuth Integration

To fetch the user information from external SSO provider, application need to be registered as a client in IDCS. Below steps providers details on registering the application in IDCS.

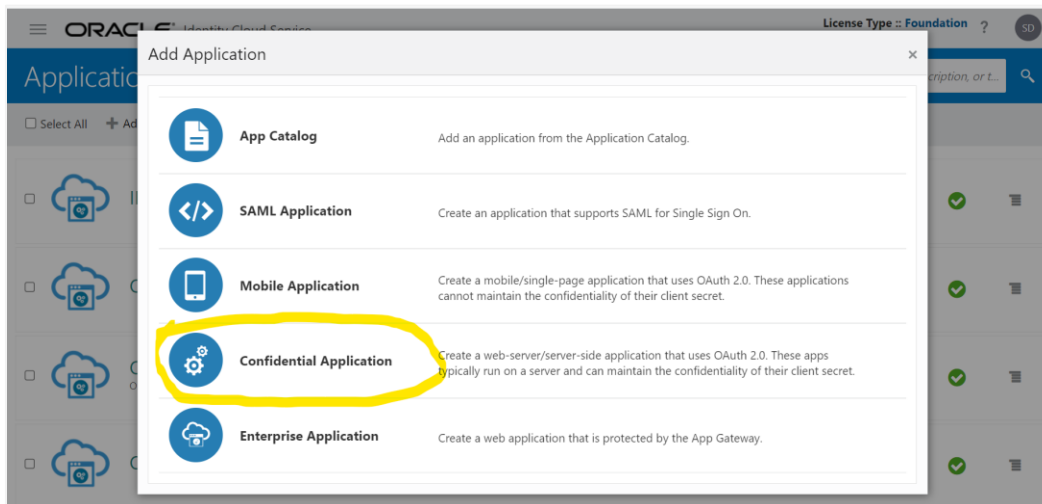
- Login to Oracle Identity Cloud Service (IDCS) console with admin login. In dashboard click on Add Application in Application and Services widget or navigate through the breadcrumb menu as highlighted in screenshot.

Dashboard



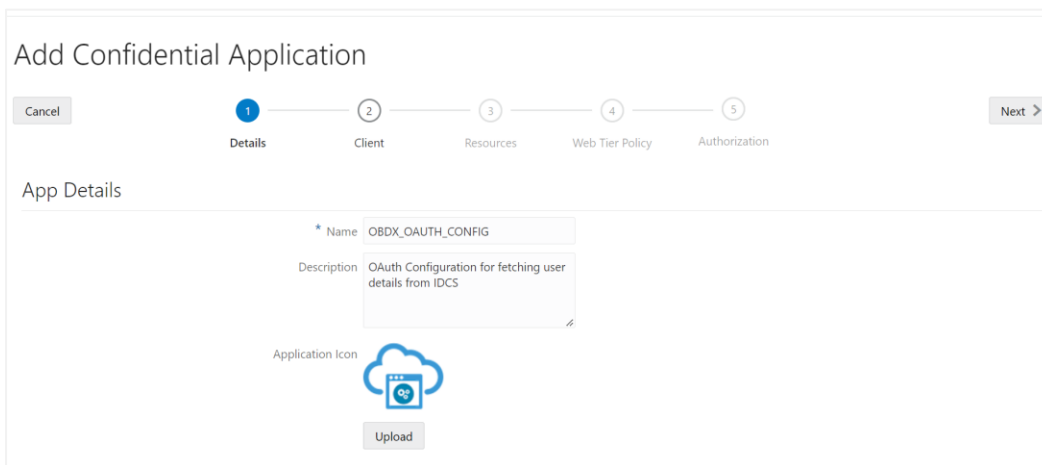
- In popup window select **Confidential Application**.

Add Application



- In **Add Confidential Application** page provide below mentioned fields and click on **Next**.
 - Name
 - Description

Add Confidential Application



- Select **Configure this application as a client now** option in screen as shown in below screenshot.

Add Confidential Application

The screenshot shows the 'Add Confidential Application' wizard at the 'Client' step. A progress bar at the top indicates steps: 1 (Details), 2 (Client), 3 (Resources), 4 (Web Tier Policy), and 5 (Authorization). Below the progress bar, there are two radio buttons: 'Configure this application as a client now' (selected and circled in yellow) and 'Skip for later'. The 'Authorization' section is expanded, showing options for 'Allowed Grant Types' (Resource Owner, Client Credentials, JWT Assertion, SAML2 Assertion, Refresh Token, Authorization Code, Implicit, Device Code, TLS Client Authentication) and 'Allow non-HTTPS URLs'. There are input fields for 'Redirect URL', 'Logout URL', and 'Post Logout Redirect URL'. At the bottom, there is a 'Security' section with 'Trusted Client Certificate' and an 'Import' button.

5. Fill below mentioned fields as per section.

i. Authorization

a. Allowed Grant Types:- Select checkbox as “Client Credentials” and “JWT Assertion”

Add Confidential Application

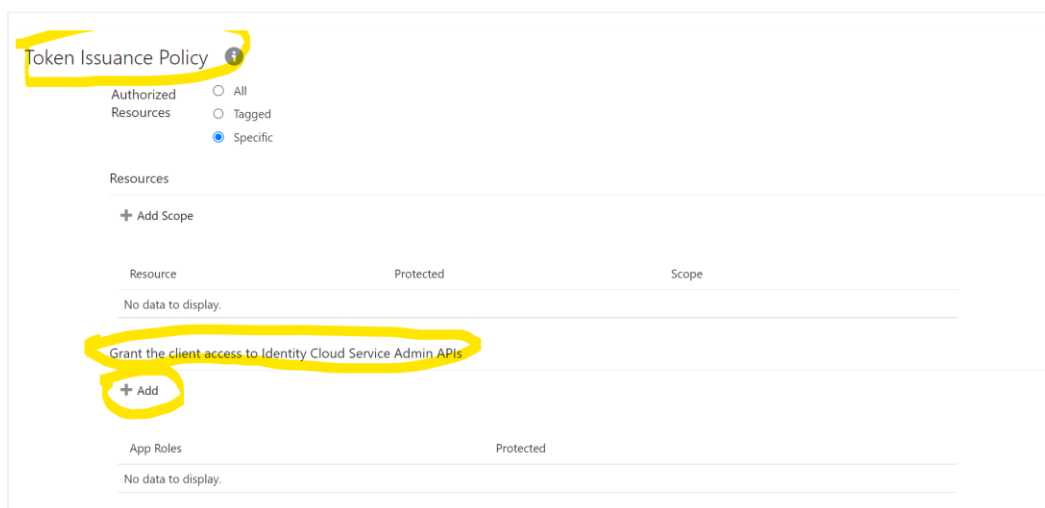
This screenshot shows the 'Add Confidential Application' wizard at the 'Client' step, similar to the previous one. The 'Configure this application as a client now' radio button is selected. The 'Authorization' section is circled in yellow. In the 'Allowed Grant Types' section, the 'Client Credentials' and 'JWT Assertion' checkboxes are selected and circled in yellow. Other options include 'Resource Owner', 'SAML2 Assertion', 'Refresh Token', 'Authorization Code', 'Implicit', 'Device Code', and 'TLS Client Authentication'. The 'Allow non-HTTPS URLs' section is also visible with input fields for 'Redirect URL', 'Logout URL', and 'Post Logout Redirect URL'. The 'Security' section at the bottom includes 'Trusted Client Certificate' and an 'Import' button. At the very bottom, there is an 'Allowed Operations' section with 'Introspect' and 'On behalf Of' options, and an 'ID Token Encryption Algorithm' dropdown set to 'None'.

ii. Token Issuance Policy

a. Authorized Resources :- Select value as “Specific”

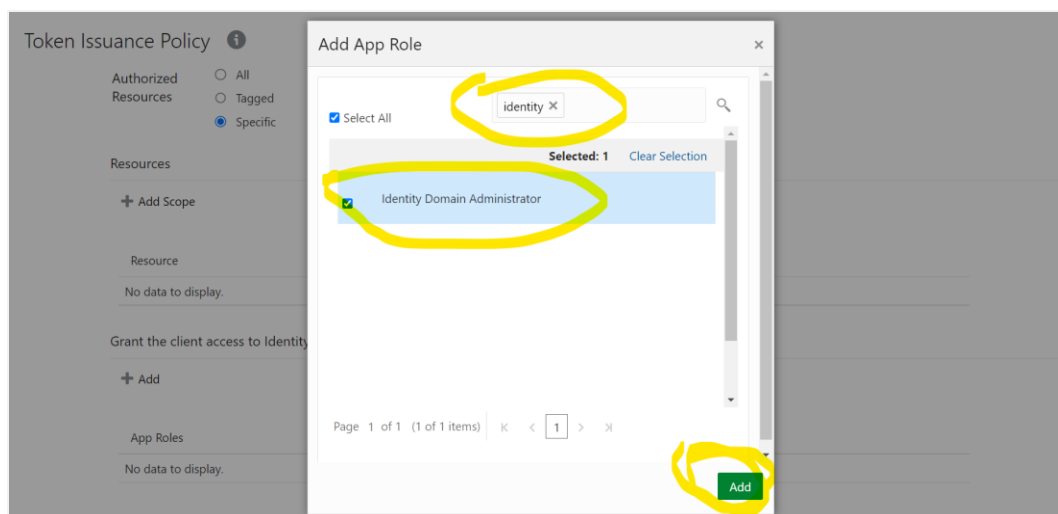
b. Grant the client access to Identity Cloud Service Admin APIs: - Click on **Add** button

Add Confidential Application



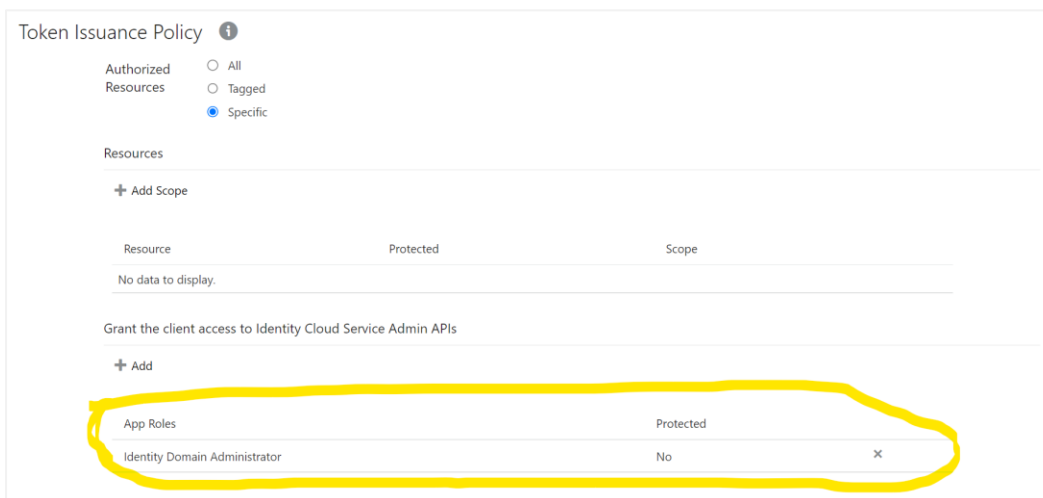
- c. In popup window search for “**Identity Domain Administrator**” and click on **Add**.

Add App Role



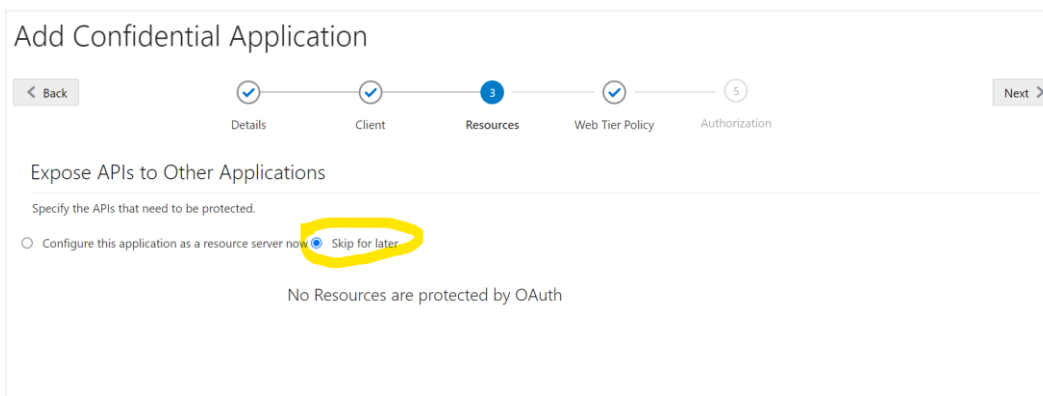
- d. Verify a row added in table for **App Roles** as shown like below screenshot

Add Confidential Application



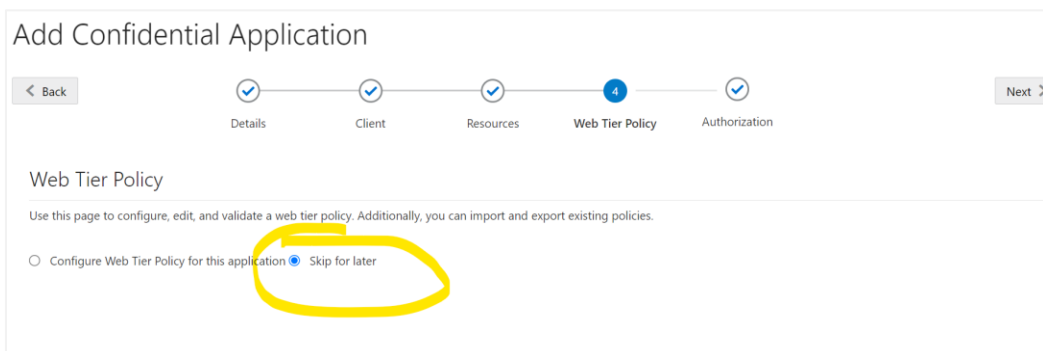
- e. Click on **Next** button on top.
- iii. Expose APIs to Other Applications: - Select “**Skip for later**” and click on **Next**.

Add Confidential Application



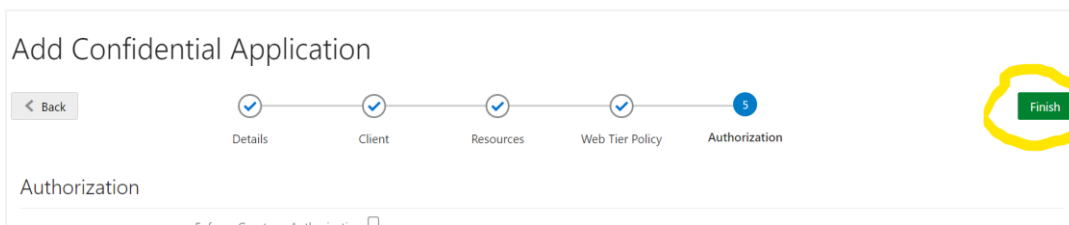
- iv. Web Tier Policy: - Select “Skip for later” and click on Next button.

Add Confidential Application



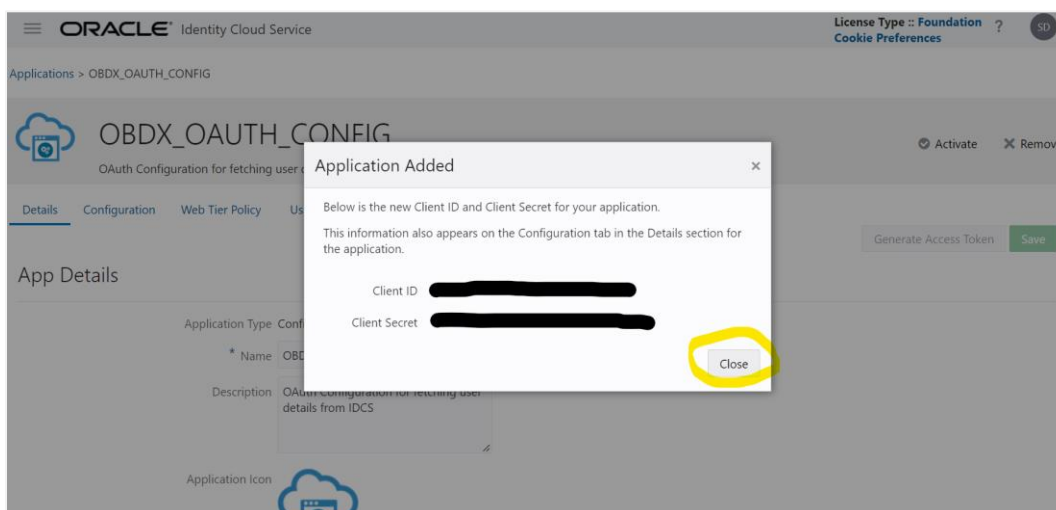
- v. Click on “Finish”

Add Confidential Application



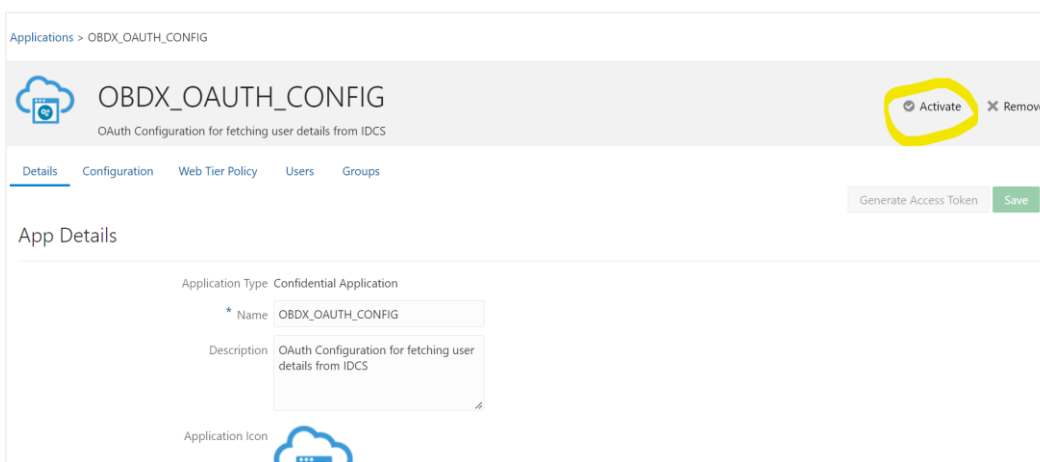
6. After finish click a popup window will open with “Client ID” and “Client Secret” as shown in below screenshot. Copy the Client Id and Client Secret to text file to keep it handy as it will be required in further steps. Once copied click on “Close”.

Add Confidential Application



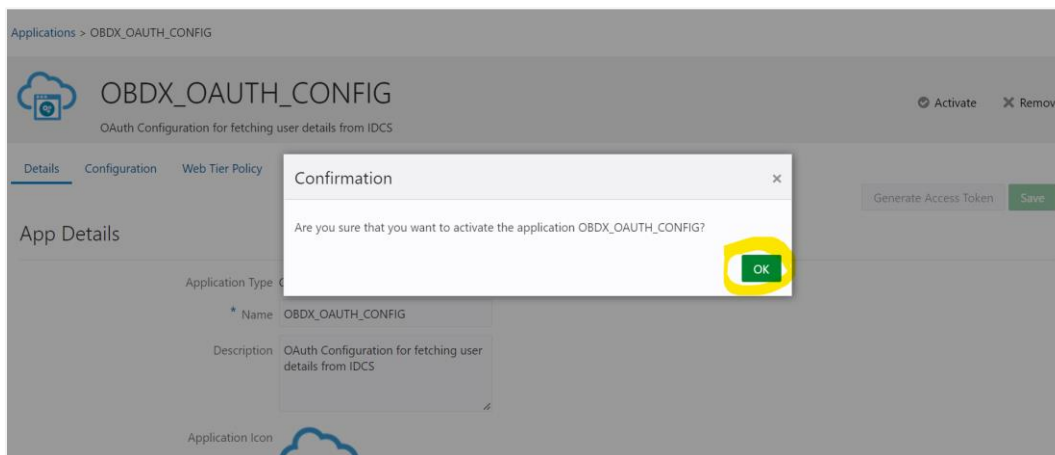
7. Click on “Activate” button to activate the application.

Edit Application



8. Popup window asking confirmation to activate the application will open, click on “Ok” to activate the application.

Edit Application



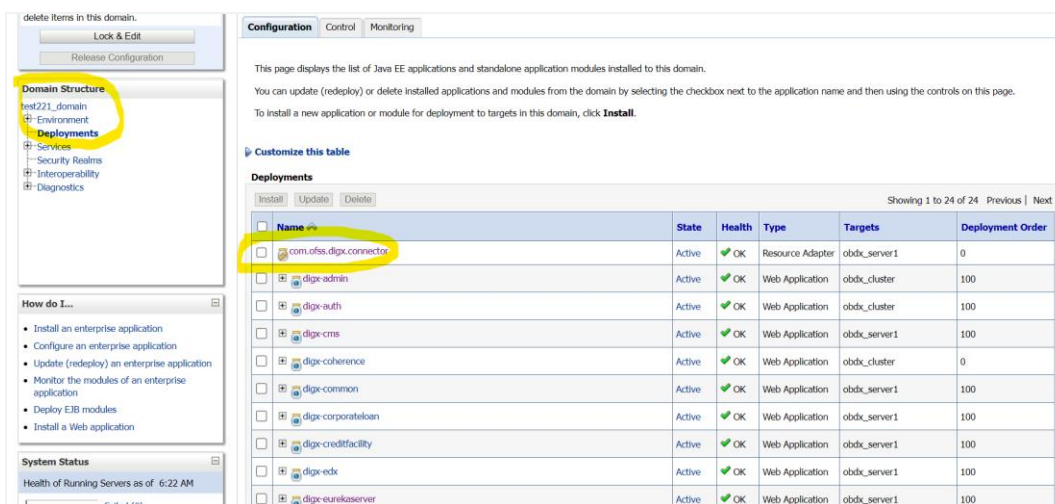
9. Logout from IDCS console.

3.7 WebLogic configuration for OAuth

To enable OAuth support on WebLogic server follow below mentioned steps.

1. Login to WebLogic console with admin login and navigate to “Domain Structure” → “Deployments”.
2. Click on “com.ofss.digx.connector”

Deployments



3. Navigate to “Configuration” → “Outbound Connection Pools” tab and click on New.

Outbound Connection Pools Configuration

The screenshot shows the Oracle WebLogic Administration Console interface. On the left is the 'Change Center' sidebar. The main content area is titled 'Settings for com.ofss.digx.connector'. The 'Configuration' tab is selected and highlighted in yellow. Below it, the 'Outbound Connection Pools' sub-tab is also highlighted in yellow. A table titled 'Outbound Connection Pool Configuration Table' is displayed, showing a single entry for the group 'javax.resource.cci.ConnectionFactory' and its 'Connection Factory Interface'.

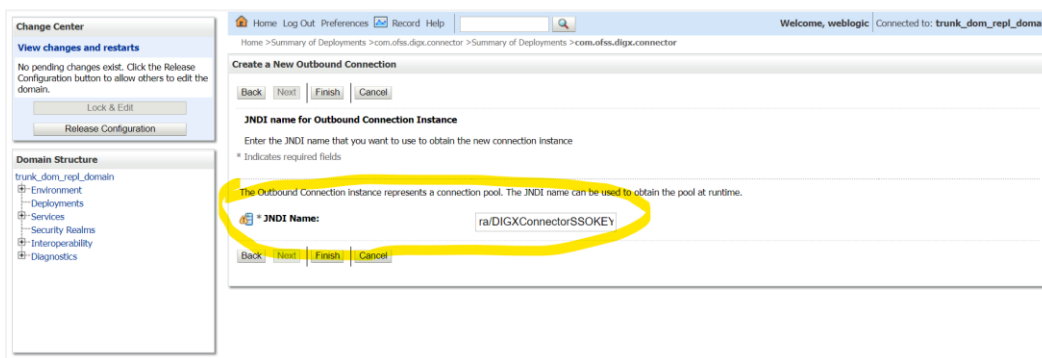
4. Select “javax.resource.cci.ConnectionFactory” and click on Next.

Outbound Connection Groups Configuration

The screenshot shows the 'Create a New Outbound Connection' wizard in the Oracle WebLogic Administration Console. The 'Next' button is highlighted in yellow. Below it, the 'Outbound Connection Groups' section is highlighted in yellow, showing a list with 'javax.resource.cci.ConnectionFactory' selected. The wizard includes navigation buttons like 'Back', 'Next', 'Finish', and 'Cancel'.

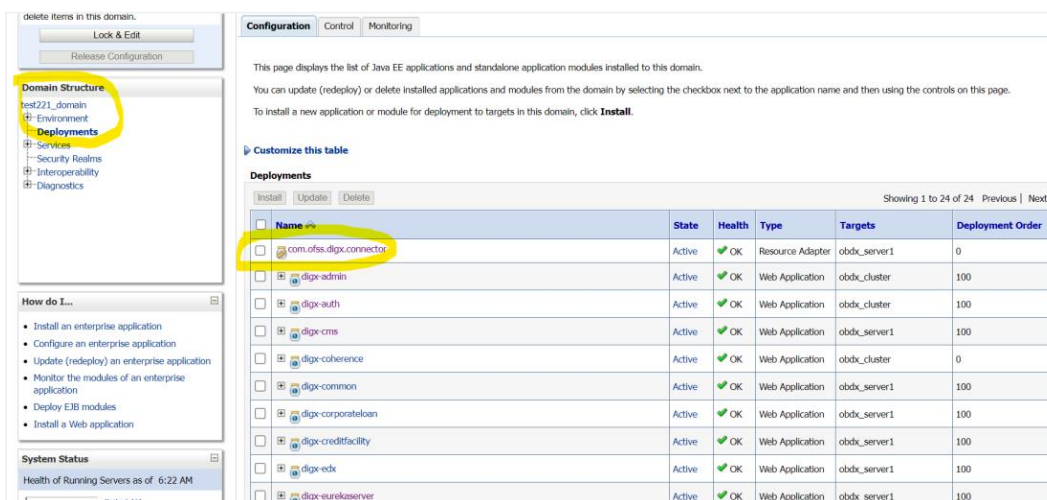
5. Enter JNDI name as ra/DIGXConnectorSSOKEY and click on Finish.

JNDI Configuration for Outbound Connection



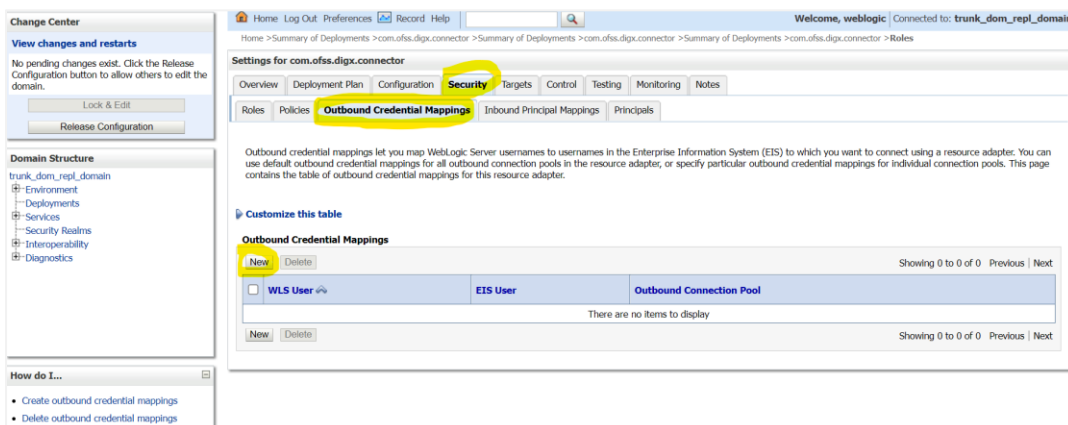
6. Again navigate to “Domain Structure” → “Deployments”.
7. Click on “com.ofss.digx.connector”.

Deployments



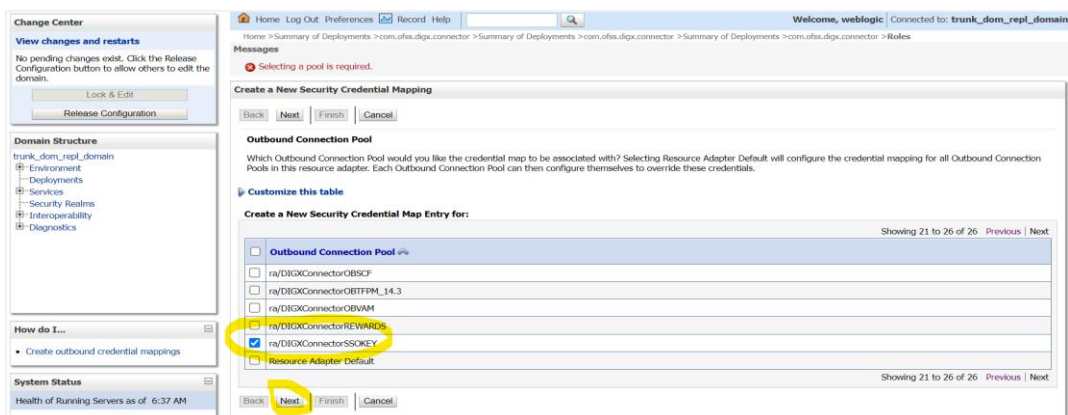
8. Navigate to “Security” → “Outbound Credentials Mapping” tab and click on New.

Outbound Credentials Mappings



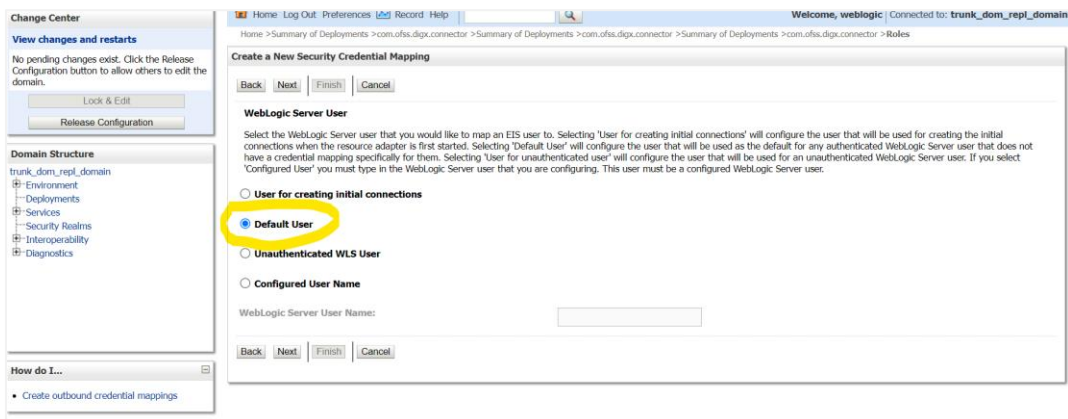
9. Select “ra/DIGXConnectorSSOKEY” by navigating using next button. Once selected as shown in below screenshot, click on Next.

Create New Security Credentials Mappings



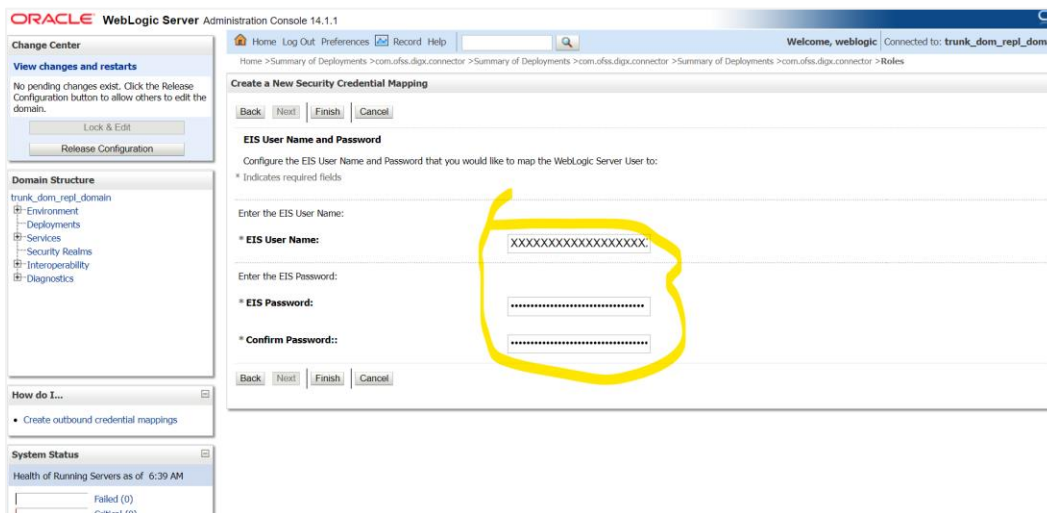
10. Select “Default User” and click on Next.

Create New Security Credentials Mappings



11. Provide the below mentioned field values as given below.
 - i. EIS User Name: - Client ID save in txt file generated from IDCS in section 3.5, step 6.
 - ii. EIS Password: - Client Secret save in txt file generated from IDCS in section 3.5, step 6.
 - iii. EIS User Name: - Client Secret save in txt file generated from IDCS section 3.5, step 6.

Configure EIS UIS Username / Password



12. Click on Finish to save the configuration.

3.8 OBAPI configuration for OAuth

To enable IDCS out of the box support for OAuth follow below mentioned steps.

1. Execute below queries on OBAPI DB.

update AUTH_CONFIG set prop_value = <IDCS_URL> where prop_id = 'IDCS_URL';

Replace <IDCS_URL> with respective url

For configuring any other service provider following queries needs to be executed.

update AUTH_CONFIG set prop_value = <SSO_PROVIDER> where prop_id = 'SSO_PROVIDER';

Replace {SSO_PROVIDER} with the one used in above query and execute the below inserts -

Insert into AUTH_CONFIG (PROP_ID, CATEGORY_ID, PROP_VALUE, FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS, OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)

```
values
({SSO_PROVIDER}_URL,'ExternalAuthenticationConfig',<PROVIDER_URL>,'N',null,'SSO
provider url','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'Y','ExternalAuthenticationConfig');
```

```
Insert into AUTH_CONFIG (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)
```

```
values ({SSO_PROVIDER}_TOKEN_PATH,'ExternalAuthenticationConfig',<TOKEN
PATH>,'N',null,'token
path','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'Y','ExternalAuthenticationConfig');
```

```
Insert into AUTH_CONFIG (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)
```

```
values ({SSO_PROVIDER}_USER_READ_PATH,'ExternalAuthenticationConfig',<USER READ
PATH>,'N',null,'user read
path','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'Y','ExternalAuthenticationConfig');
```

```
Insert into AUTH_CONFIG (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)
```

```
values
({SSO_PROVIDER}_USER_READ_ASSEMBLER,'ExternalAuthenticationConfig',<FULLY
QUALIFIED NAME OF ASSEMBLER>,'N',null,'user read
assembler','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'Y','ExternalAuthenticationConfig');
```

```
Insert into AUTH_CONFIG (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)
```

```
values
({SSO_PROVIDER}_TOKEN_SCOPE,'ExternalAuthenticationConfig',<SCOPE>,'N',null,'Client
credentials token
scope','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'Y','ExternalAuthenticationConfig');
```

- Restart all the servers in domain.

3.9 **Default Admin Configuration**

OBAPI installer comes pre-shipped admin user with name “superadmin”,so in order to login into the OBAPI application for completing Day 1 maintenances the same user need to be created in SSO Provider with same name post SSO integration.

3.10 Logout Configurations

Below query needs to be executed as part of the logout configurations.

```
Insert into DIGX_FW_CONFIG_ALL_B
(PROP_ID,CATEGORY_ID,PROP_VALUE,FACTORY_SHIPPED_FLAG,PROP_COMMENTS,S
UMMARY_TEXT,CREATED_BY,CREATION_DATE,LAST_UPDATED_BY,LAST_UPDATED_D
ATE,OBJECT_STATUS,OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTOR)
values ('SSO_LOGOUT_URL','dayoneconfig','<LOGOUT_URL>','Y',null,'SSO logout
Url','ofssuser',sysdate,'ofssuser',sysdate,'A',1,'N',null);
```

Replace <LOGOUT_URL> with respective url.

[Home](#)