Steps to setup Oracle Connector

1. Clone the repository

Clone the oracle connector repository using following command:
git clone https://bitbucket.org/kilroy/oracle-connector.git

2. Configure Environment file

Here is the sample environment file:

```
OBC_ADMIN_HOST=https://idcs-a0e6xxxxxxxxxxxxxxxxxxxxxxxx4daa988.identity.oraclecloud.com #Get the OBC_ADMIN_HOST from Oracle Cloud
AUTHORIZATION_TOKEN=Uhx0el9iKHxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxkG5bB1QDHMm #Get the AUTHORIZATION_TOKEN from Oracle Cloud
OBC_ADMIN_USERNAME=TestUser #Use Oracle Cloud Admin Username
OBC_ADMIN_PASSWORD=Test@User123 #Use Oracle Cloud Admin Password
OBC_ADMIN_SCOPE=urn:opc:idm:__myscopes__
REDIS_HOST=obc-redis-cache
REDIS_PORT=6379
APP_PORT=3000
PORT=5818
API_AUTHORIZATION_TOKEN=b3JhY2xlLWJjLWNvbm5lY3Rvci1hcGkta2V5Ojk3ZWVxMmFjLTgxajItODU3My0wZTFlZS0zNXA5NDQxaTFrYWNk
```

To configure the above environment file, the env variable highlighted with color red needs to be replaced with your oracle cloud details. The steps to get those information are explained below:

Each environment variables are explained below:
Firstly you need to login to the oracle cloud from here: https://cloud.oracle.com/

1. OBC_ADMIN_HOST

Search for federation on oracle cloud. Then click on Federation (highlighted with the red box) as shown below:

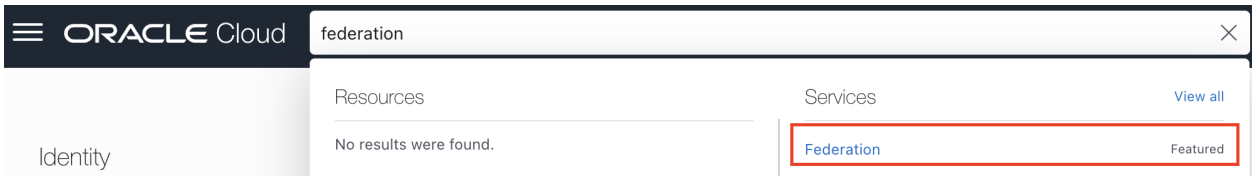


Fig. 1.1

Then click on the federation service available. In our case you have OracleIdentityCloudService as shown below. The name of the federation will be different in your case.

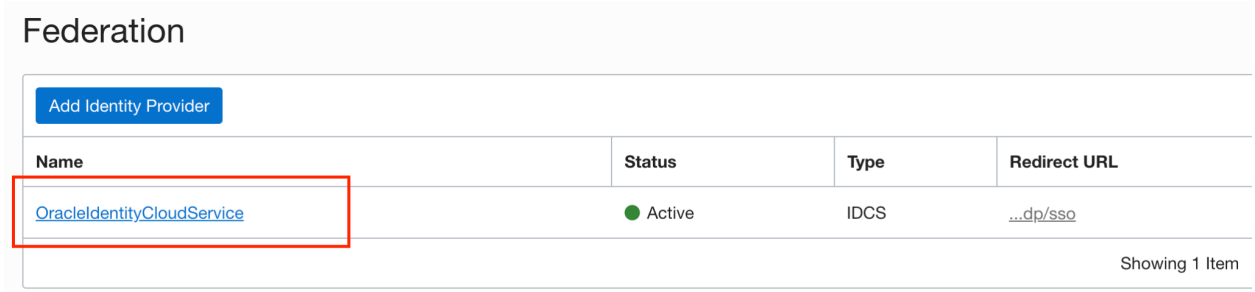


Fig. 1.2

It will open this window where you can find the Oracle Identity Cloud Service Console. You need to copy the URL shown on the red box only up to the text ..oralcecloud.com. And paste on the env file variable OBC_ADMIN_HOST as shown on the env file above. The full URL will be https://idcs-e2f496a4xxxxxxxxxe90dac0af081.identity.oraclecloud.com

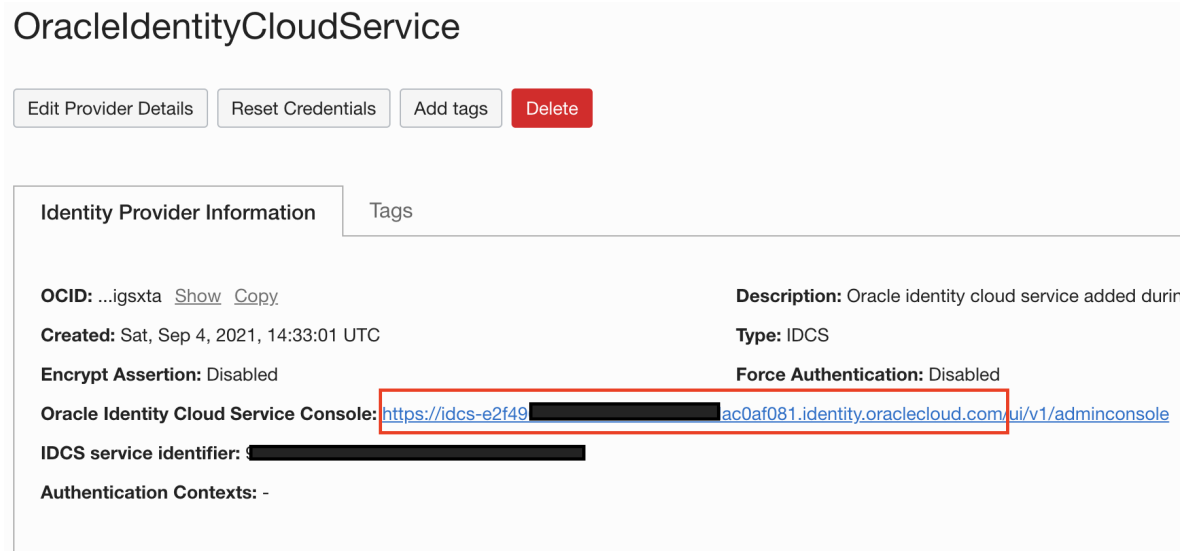
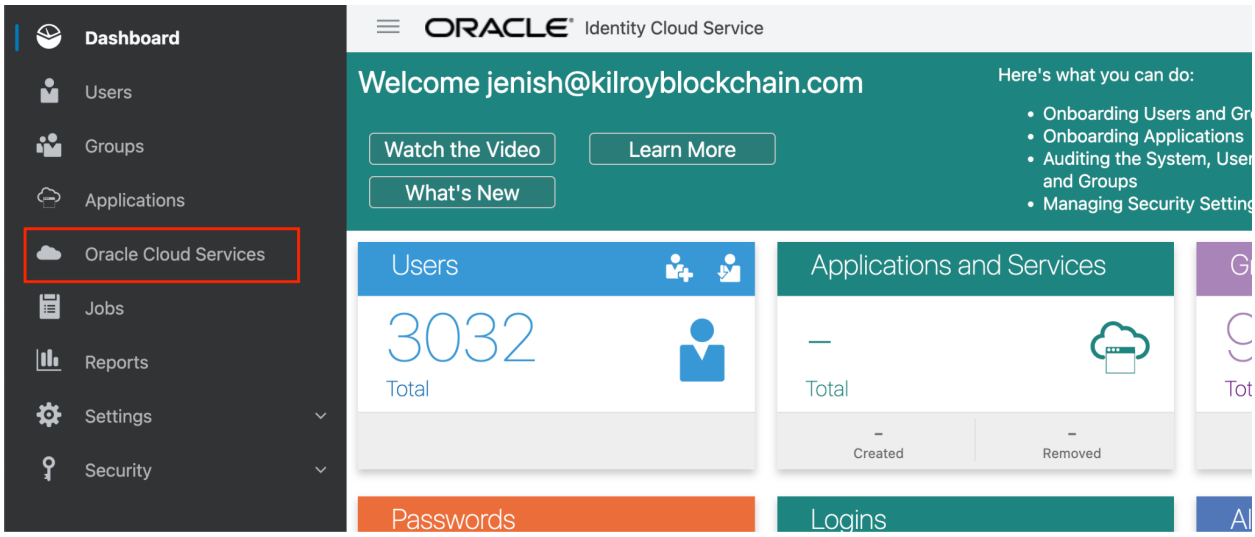


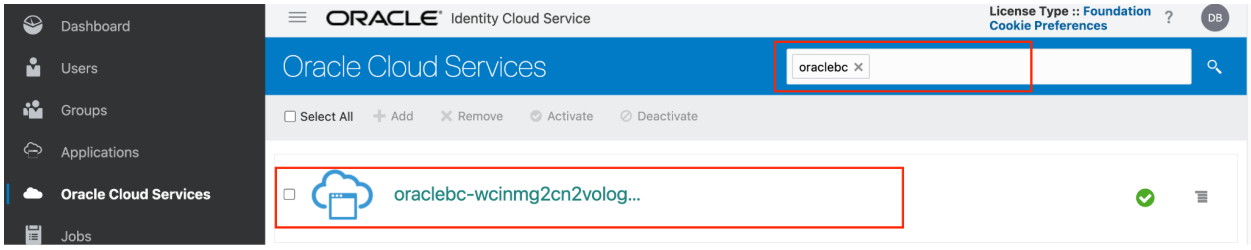
Fig. 1.3

2. AUTHORIZATION_TOKEN
   You assume that you are still in the same window as shown on Fig.1.3. Now you click on the URL https://idcs-e2f496a4xxxxxxxxxe90dac0af081.identity.oraclecloud.com as shown on Fig. 1.3. This will open a new tab on your browser.

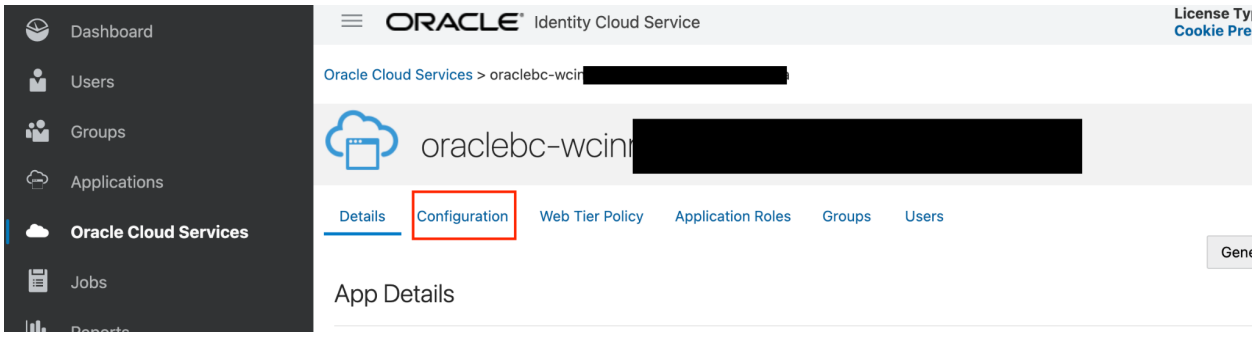
   Then you navigate to Oracle Cloud Service on the sidebar.



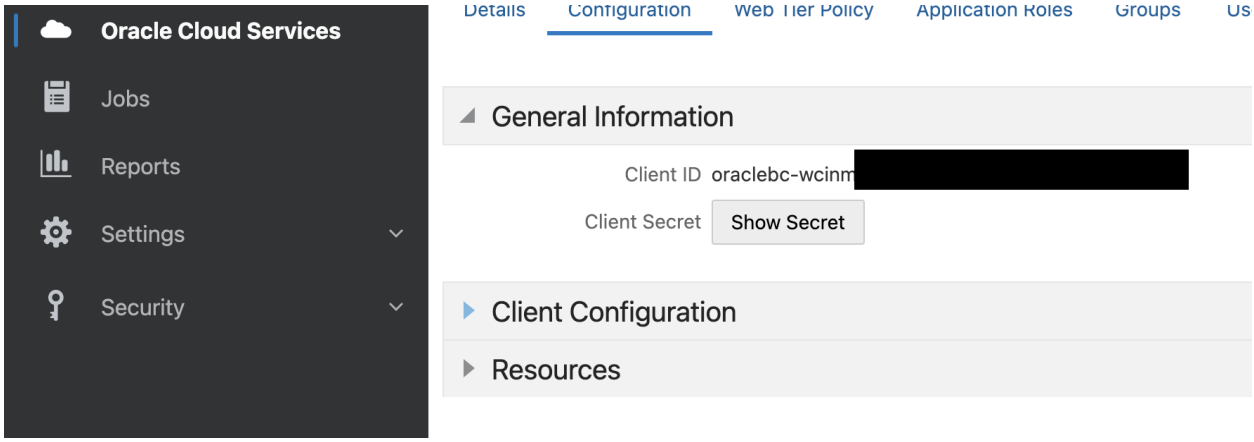
Now search for the oracle cloud service. In our case you have oraclebc-xxxx cloud service. You need to add a new Oracle Cloud Service if you do not have any.



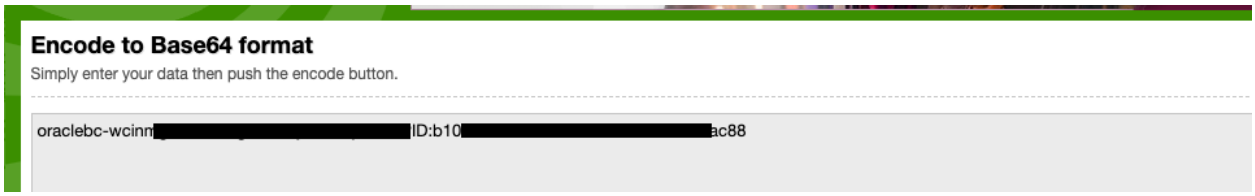
It will show the following window. Click on the Configuration tab next to Details.



Now we copy the Client ID and Client Secret from the General Information. Client Secret can be copied when clicked on the button **Show Secret.**



After copying the Client ID and Client Secret from this, you need to encode it to base64 string. To do that you can click on the link https://www.base64encode.org/ . Here copy the client id and client secret in the format <client_id>:<client_secret>. You can see the example below:
oraclebc-wcinxxxxxxxxxxxxxxxxxxxxID:b10xxxx-xxx-xxxxxxx-xxxxx88



Then you can click on the encode button like below. You can see the encoded base64 string. You copy that string and paste on the env file variable AUTHORIZATION_TOKEN as shown on the env file above.

3. OBC_ADMIN_USERNAME
   This is the username of any oracle cloud admin user. This is the username used to login on oracle cloud.

4. OBC_ADMIN_PASSWORD
   This is the password of oracle cloud admin user. This is the password used to login on oracle cloud.

5. OBC_ADMIN_SCOPE

This is the scope needed to provide while accessing the oracle cloud APIs. The scope can be default mentioned on the environment variable.

6. REDIS_HOST
   This is the redis host URL which will be the localhost in our case. The redis container will be run when you run the script for running the oracle connector. The redis here is used to store the access token when logged in to the oracle cloud from the oracle connector itself. The access token then will be used on every other operation like create user, get group list and assign user to oracle group. Leave the field default for testing.

7. REDIS_PORT
   This is the redis deployed port number. Leave the field default for testing.

8. APP_PORT
   This is the port of the oracle connector to be listened to. Leave the field default for testing.

9. API_AUTHORIZATION_TOKEN
   API Authorization token is the unique token generated for a backed application to connect to our oracle connector. The authorization token will be used later on the backend application environment file. you need to set the OC_AUHTORIZATION_TOKEN on the backend application environment file same as the current token. Leave the field default for testing.