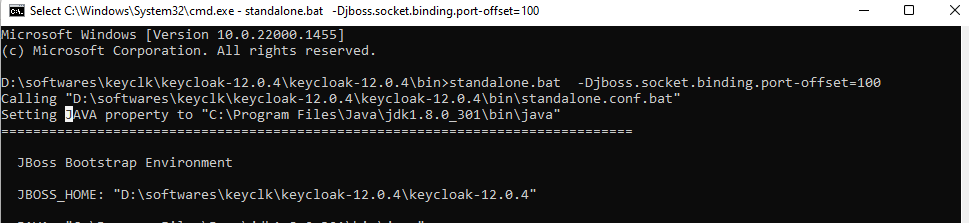
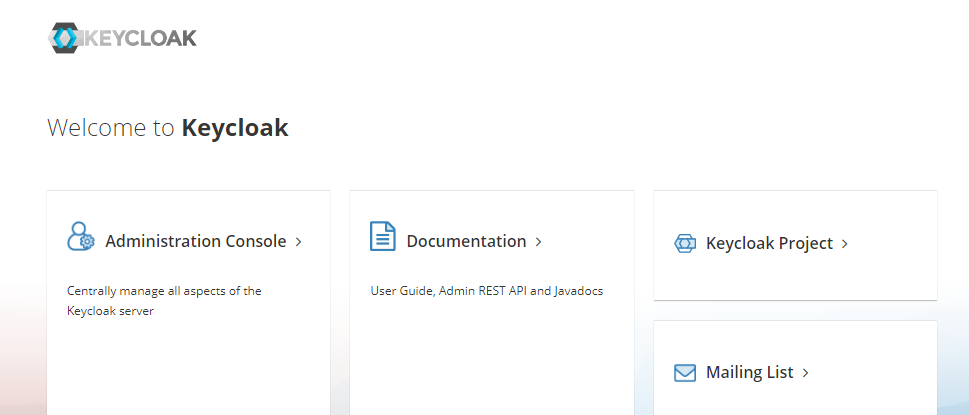
1. Download the keyCloak Identity and Access Management Tool. Download the zip file
2. Navigate to <<KeyCloack Install Directory>/bin and start the server using following command

standalone.bat -Djboss.socket.binding.port-offset=100

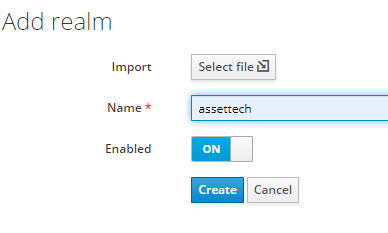
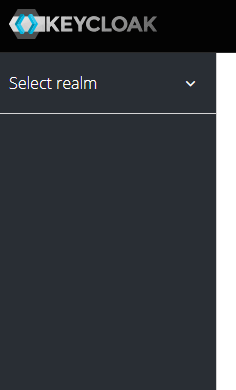


1. Access the server using the following url and click on the Administration Console. For the first time give the credentials

<http://localhost:8180/auth/>

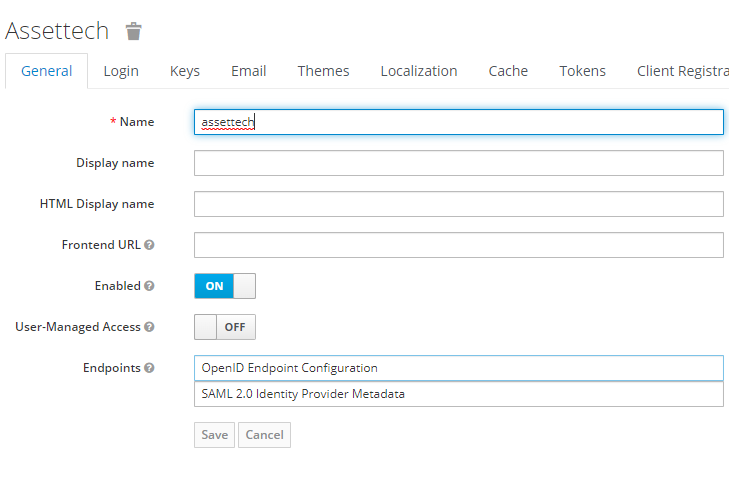


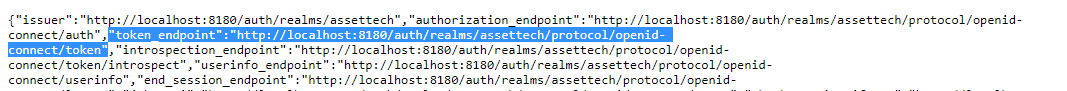
1. Select a Realm and Add Realm .As per the KeyClocak Realm is the container on which the clients, roles and scopes are created



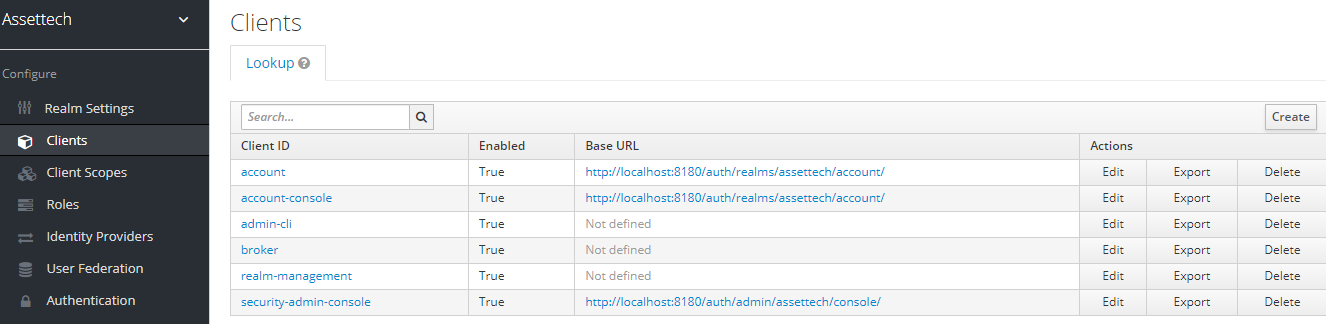
1. After creating the realm as shown in the screenshot below,click on the OpenID Endpoint Configuration and copy the tokenendpoint url. Just enter the Realm name

and other entries can be ignored

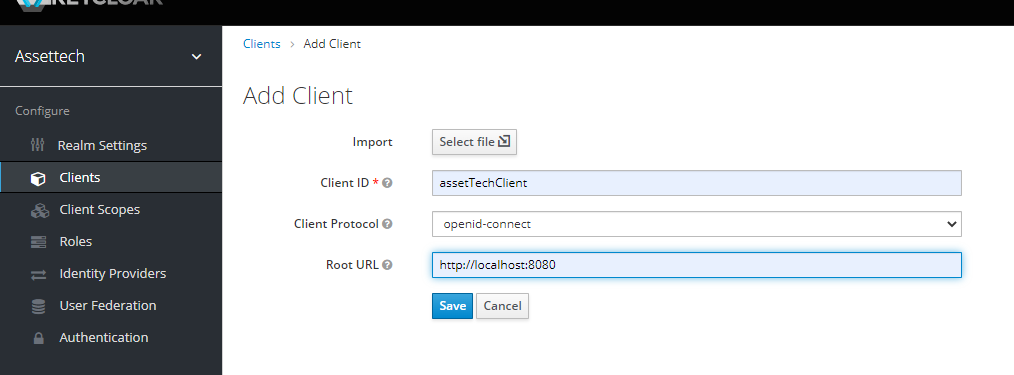




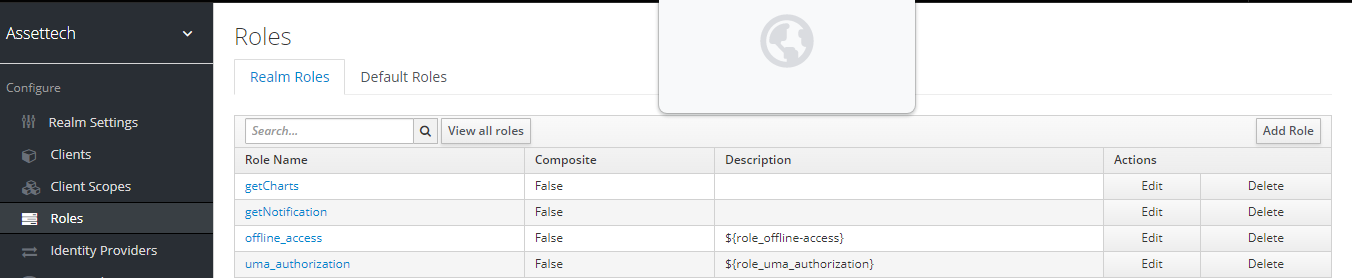
1. Then create a Client. Client is the identity of the application(Any WebClient, Postman,Application etc). For creating client click on the create button on right hand top corner



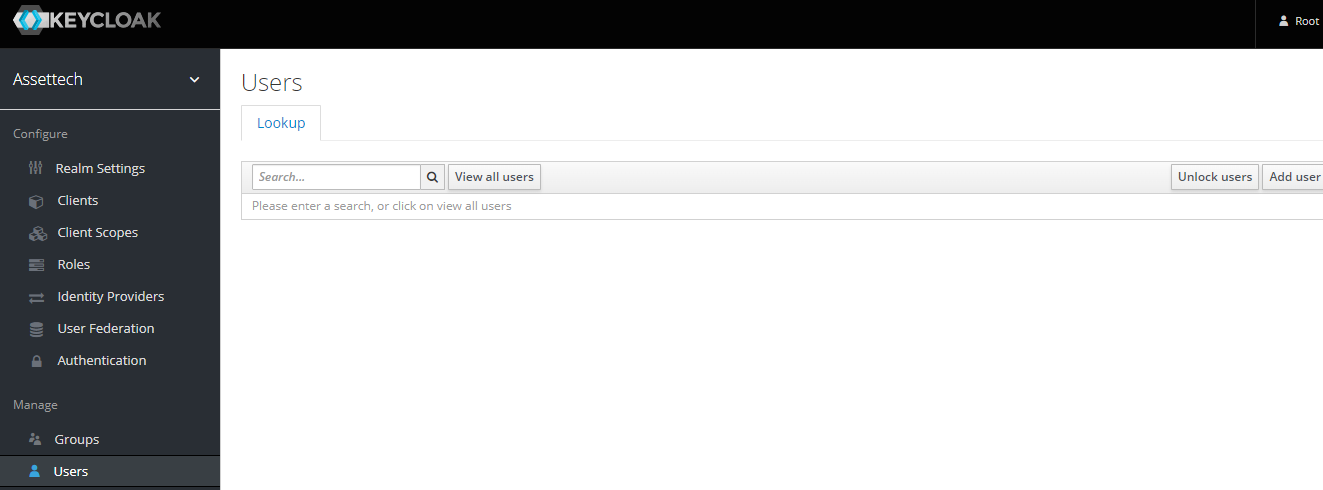
1. While adding the client give the Clientid and for Client Protocol select openid-connect. This is very important. In Neom we are not using SAML. Then give root url of the application which should be used to collect the client credentials. In our case it is http://localhost:8080. There are many parameters for client identity like Roles ,Scopes etc. Depending of the application it will be used

c

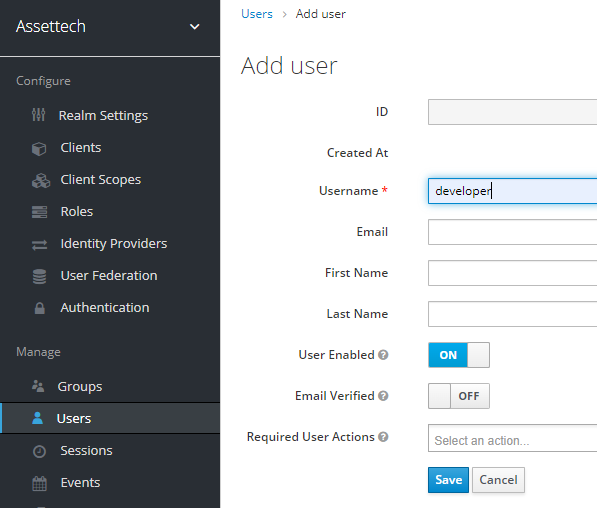
1. Create the Roles which can be associated with the users of a particular client



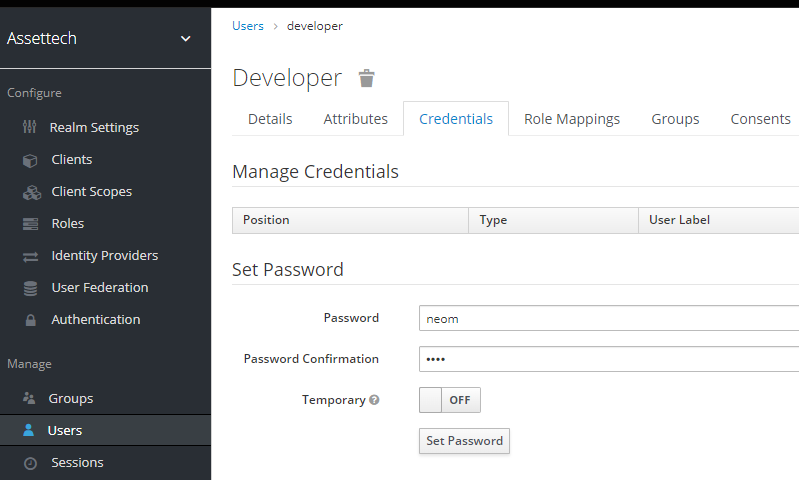
1. After a client is created, we need to use users corresponding to client. In the left panel click Users and create users in the main window right hand top corner



1. Give the user name and click on credentials tab. Give the password and note the password and make Temporary as OFF. This is required while accessing the application

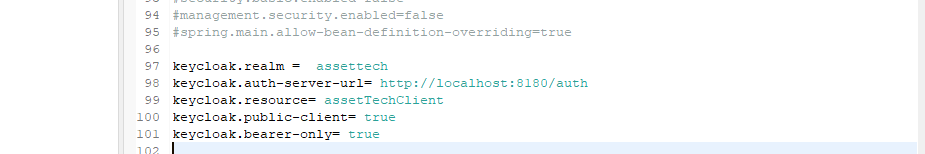


1. click on credentials tab. Give the password and note the password and make Temporary as OFF. This is required while accessing the application



**Changes in the .properties and Spring Security Configuration**

In The following entries needs to be added in Application.properties



The details needs to be given are

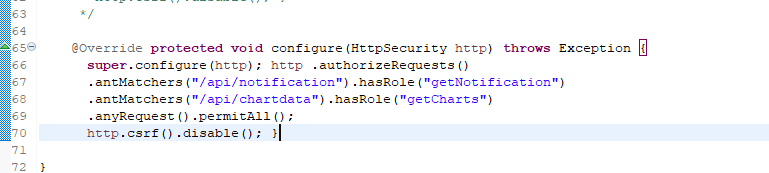
1. Realm
2. The url where the Kloack server is running
3. The Client name which we created above
4. Then other 2 entries can be defaulted to true

A new configuration needs to be created for KeyCloack in the Spring boot Application



Since we are using Spring 2.7.3 in the Spring boot version in the current MVP application, we need this configuration. ` upgrade the spring boot version, we need to change the above configuration

In the POC current implementation, we are are associating /api/notification and /api/chartdata with a Role we created in “getNotification” and “getCharts” in the KeyCloack Server above

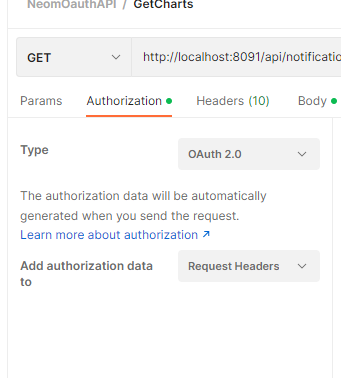


If a call is made to the above said API’s with a JWT token which doesn’t contain this role , we get forbidden exception as explained in the next section

**Accessing the protected API’s through Postman**

Setting up the Postman to generate the token

1. Under the Authroization tab in Postman, select OAuth 2.0



1. In the right hand side set the following values for each of the fields

* **Token**: Available Tokens
* **Header Prefix**: Bearer
* Under Configure New Token Heading
* **Token Name :** Any name
* **Grant Type:** This is very important. The Grant Types can be any of the following

1. Authorization Code

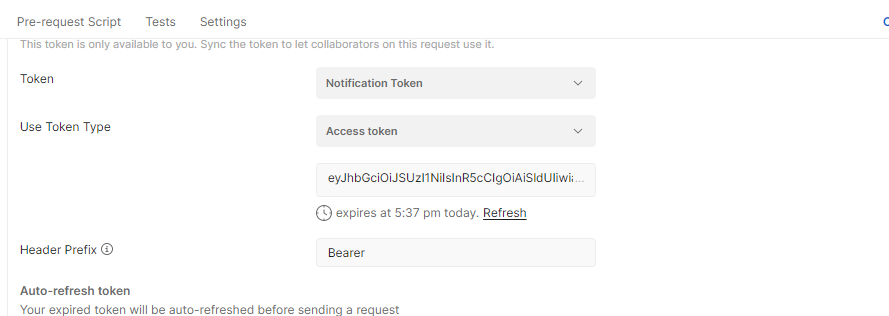
2. Implicit

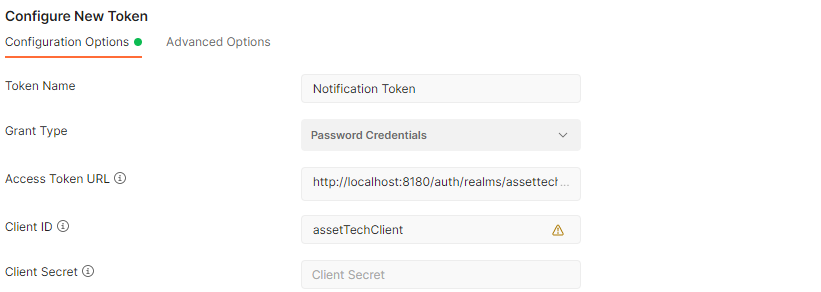
3. Password Credentials

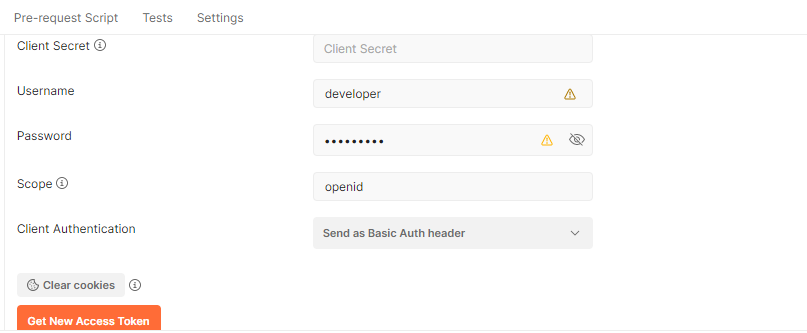
4. Client credentials

The values we need to give when generating the token depends on the Grant Type. Here we are using Password Credentials. So we need to give UserName and Password

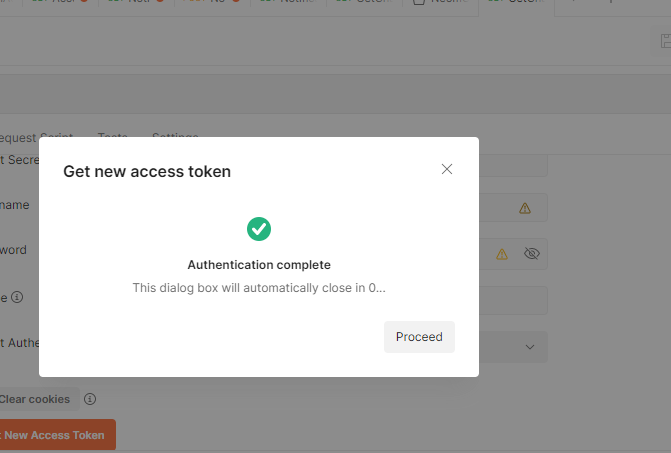
* **Access Token Url**: The value which we copied above in Step 5. For example http://localhost:8180/auth/realms/assettech/protocol/openid-connect/token
* **Client ID:** The Client name which we created in Step 7
* **Username:** The name we give while creating the User in the Step 11
* **Password:** The password is the value we give in the Credentials tab in Step 2
* **Scope:** openid
* **Client Authentication:** Send as Basic Auth header

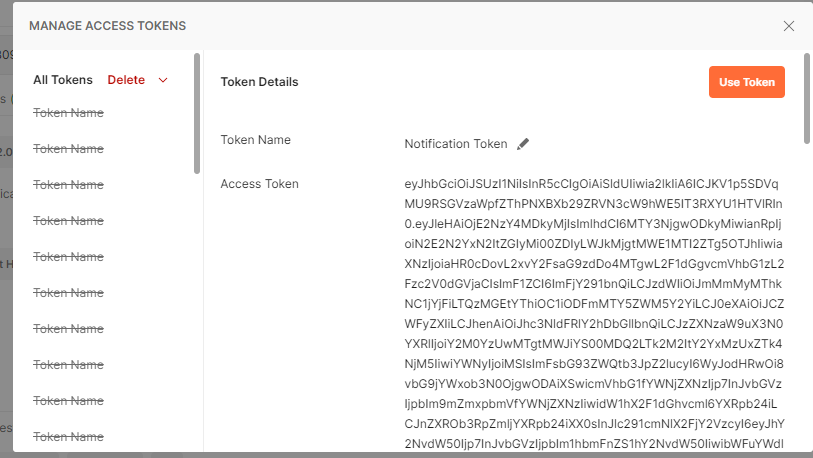




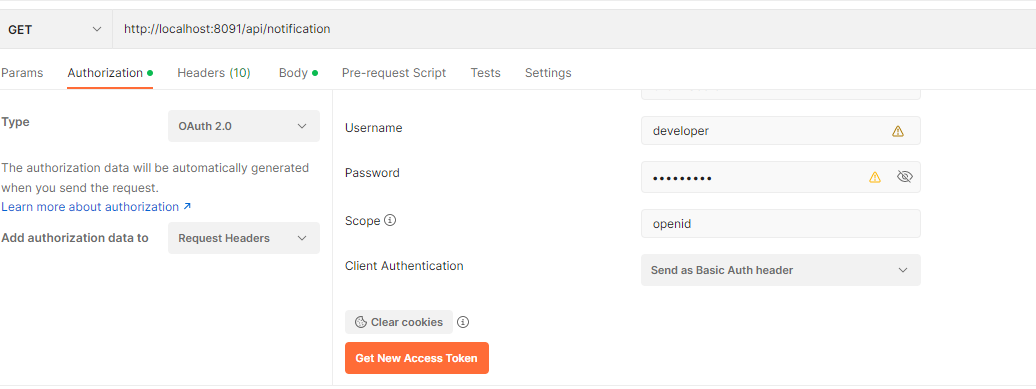


1. Click on AccessNewToken button. When we get the token, Click on proceed and Use Token. The token will be sent as header





1. Here for the developer user, I have configured getNotification Role. So if I generate token using these values, I get the valid response

\

1. Now I generate the token, using tester role, I get forbidden error

