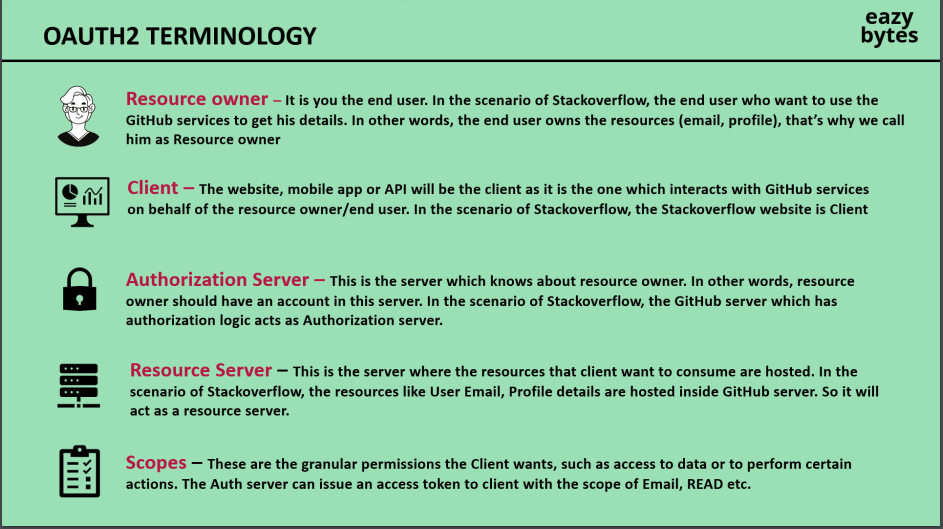
How Oauth works:

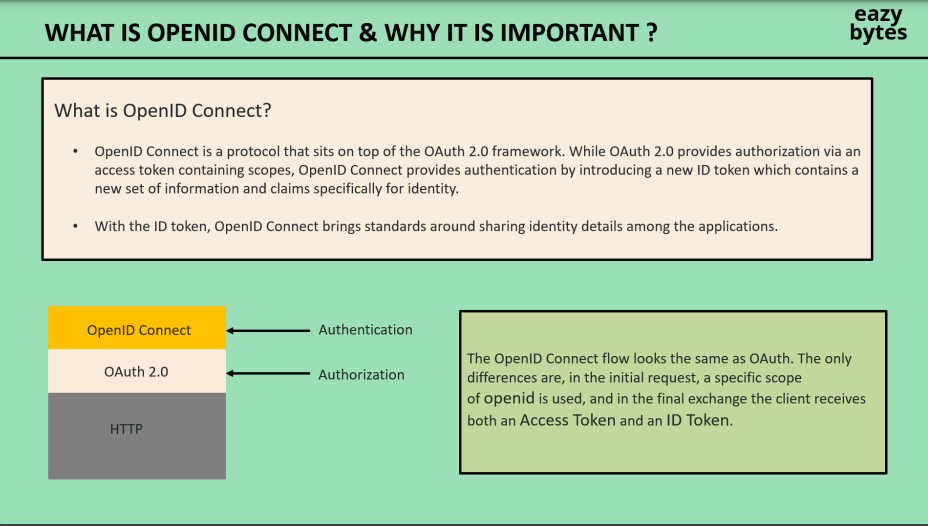
**Note:** Please read the pdf attached in the repo for more clarity about OAuth2.0

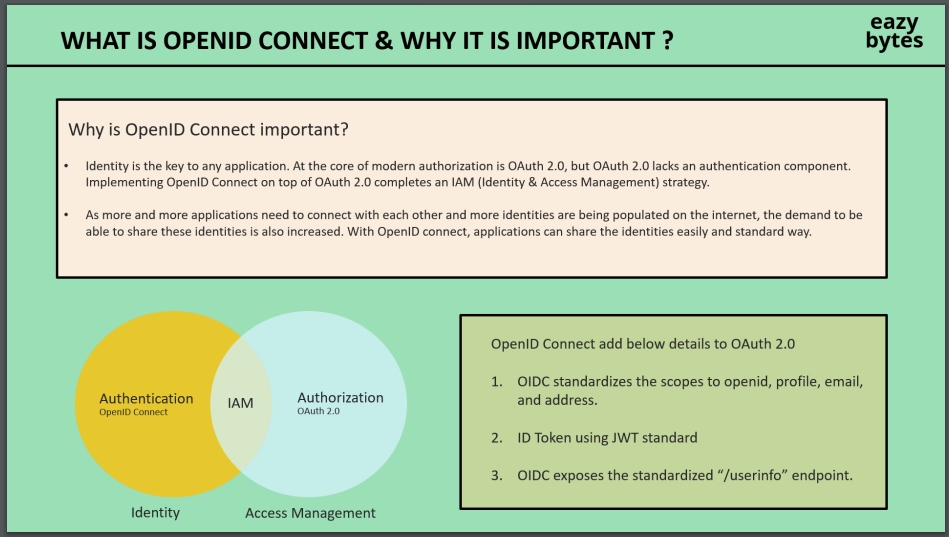
OAuth is used for Authorization but not for authentication.



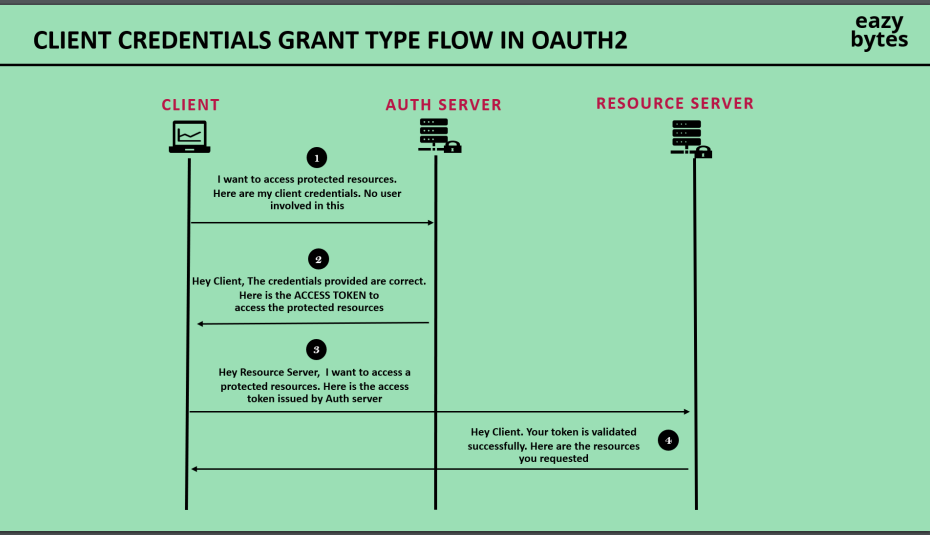
OpenID connect - is the wrapper over the Oauth2 that provide the authentication whereas Oauth2 provide the Authorization only.

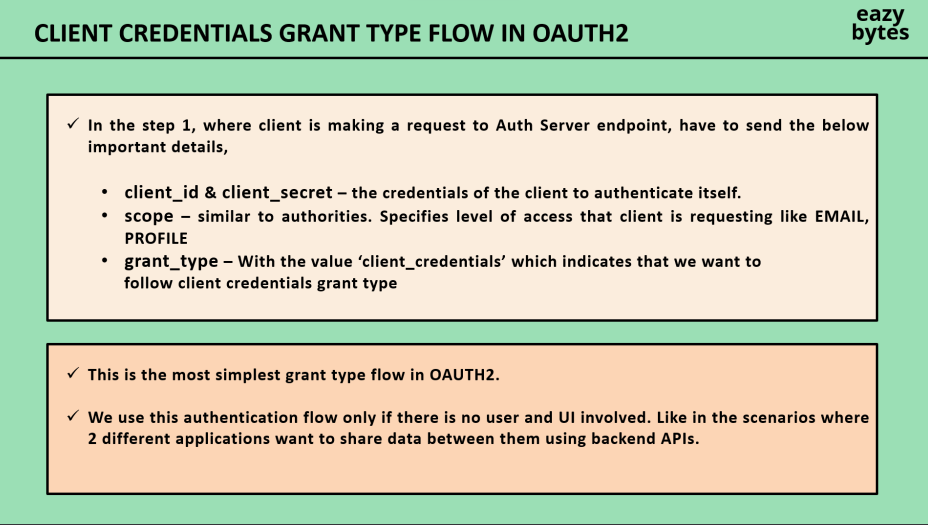
OpenId provide the ID token whereas Oath provide the access token.





Client credential grant type flow is one of the approache out of many that we are using in this project.



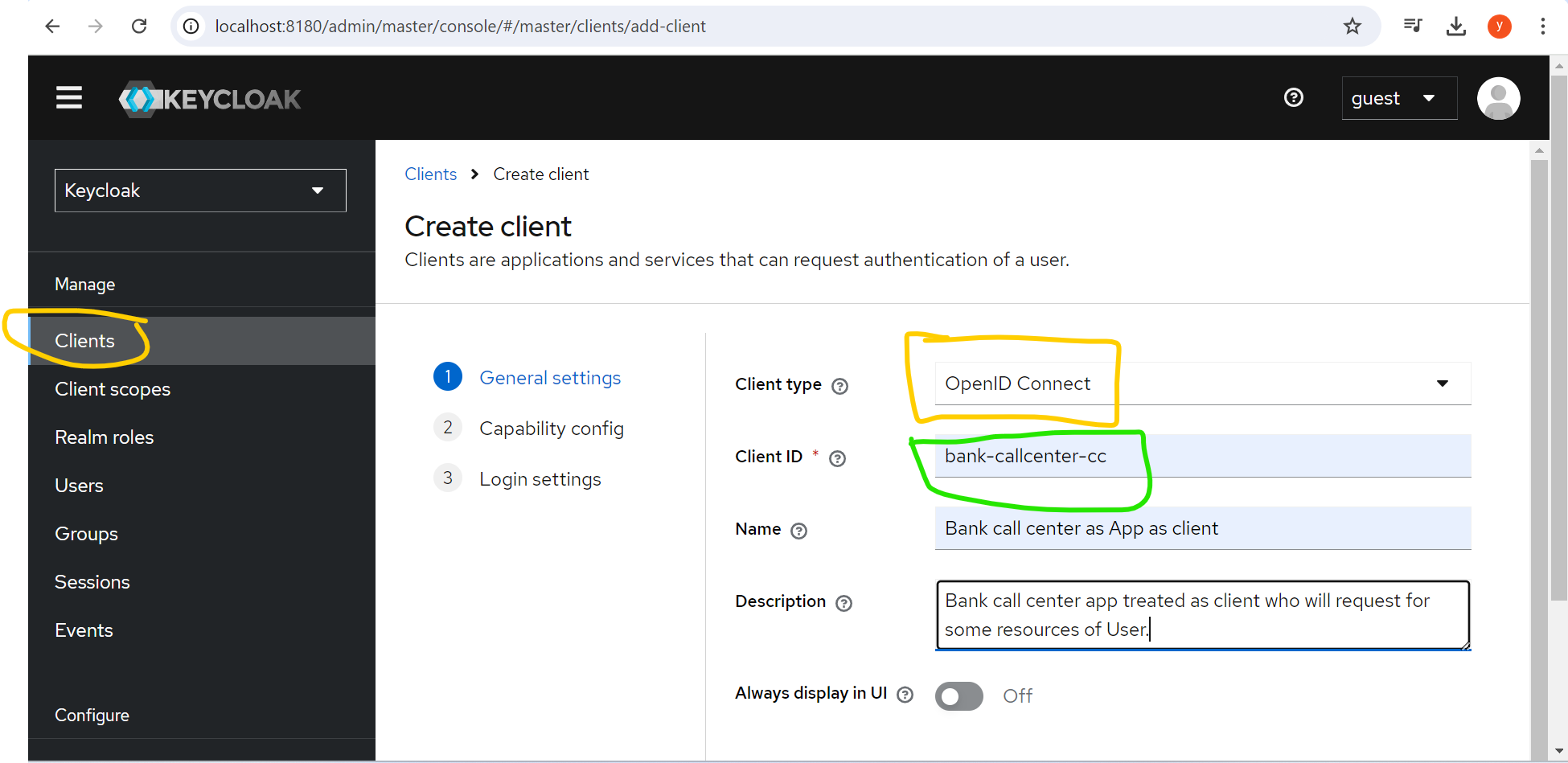


How to configure Keycloak:

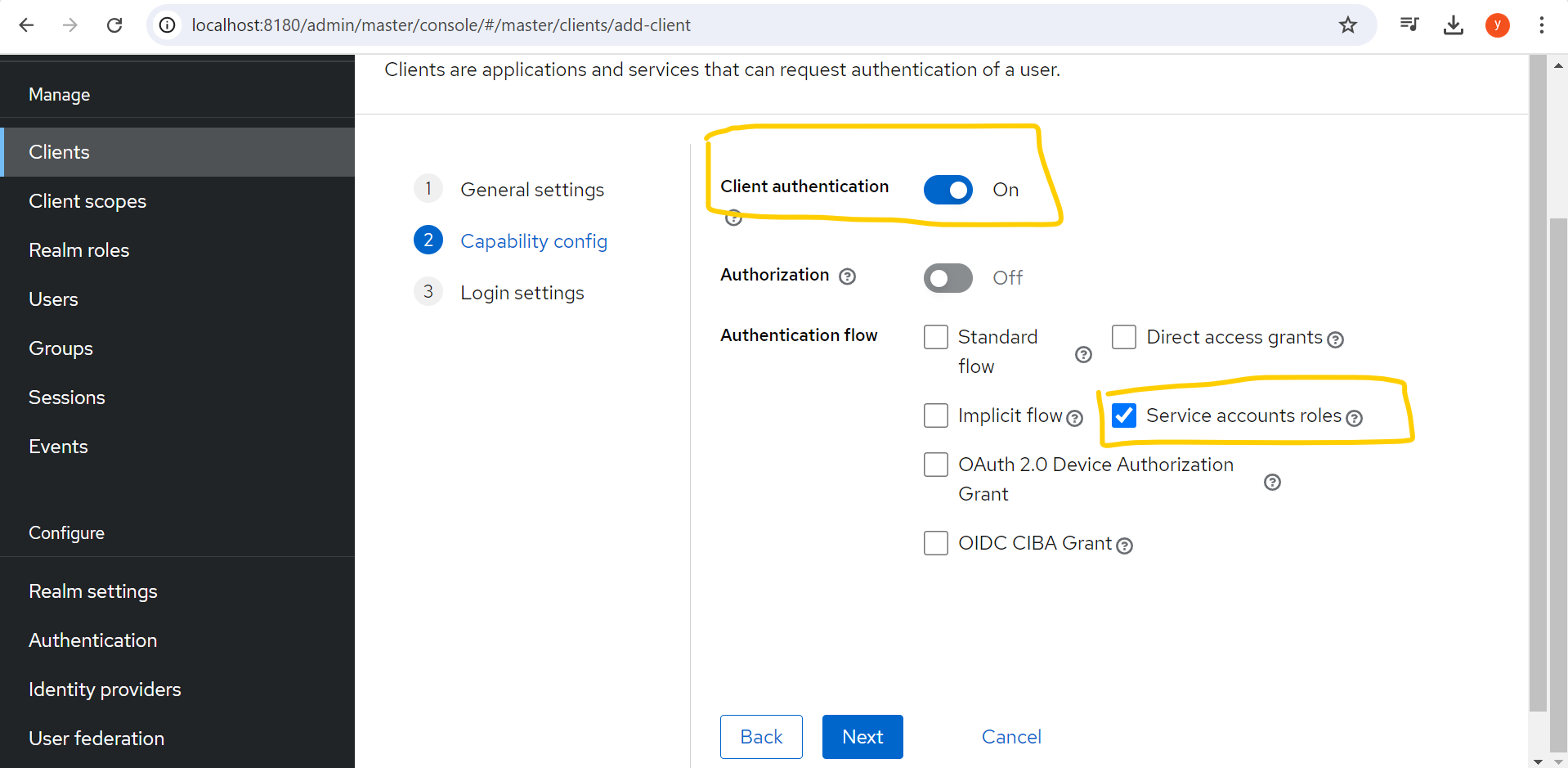
1. Download keycloak for windows
2. Run in Dev mode command : bin\kc.bat start-dev --http-port=8180
3. Hit on browser : localhost:8180
4. Signing and start

Configure Key cloak for our Bank application

1. Create(register) client in keycloak .



In this above ,we have created a clientId in green color. Press next.

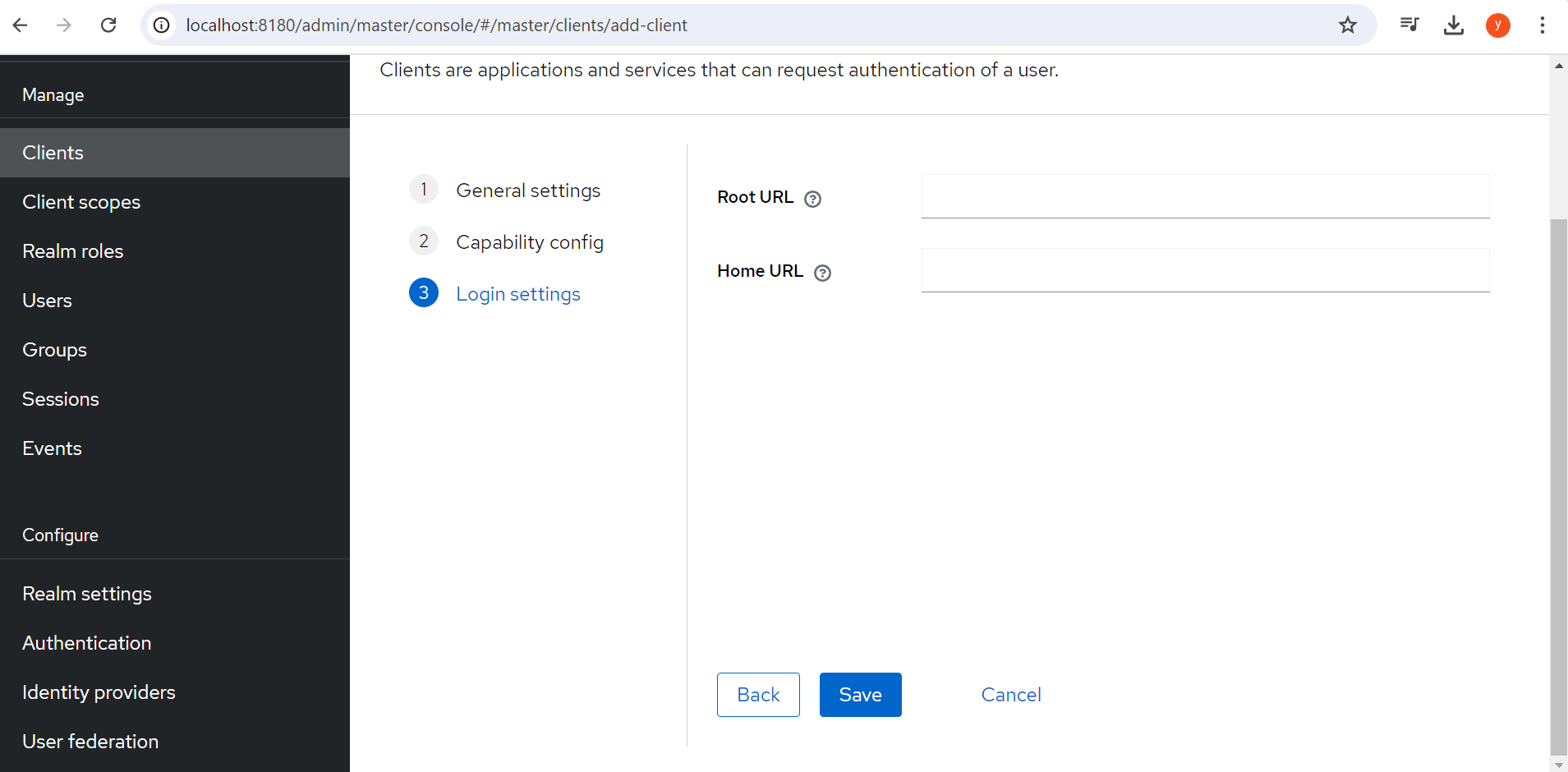


Here client authentication is enable ,because client have to provide the client credentials to access the resources, then only Oauth server provide the AccessToken.

And the service accounts roles checkbox mentioned:

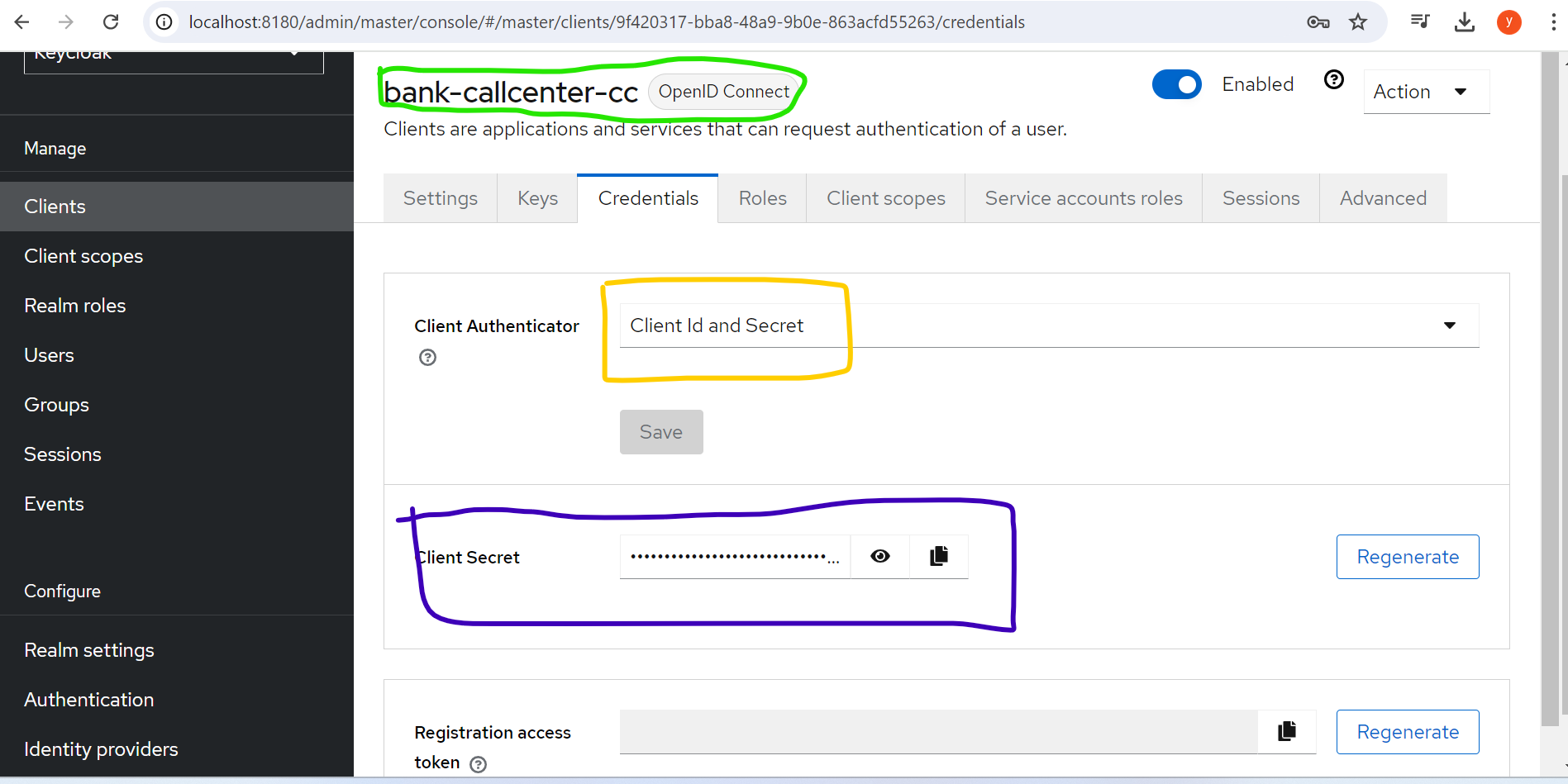
Allows you to authenticate this client to Keycloak and retrieve access token dedicated to this client. In terms of OAuth2 specification, this enables support of 'Client Credentials Grant' for this client.

Press next



And press save.

Now you can check your client credentials:



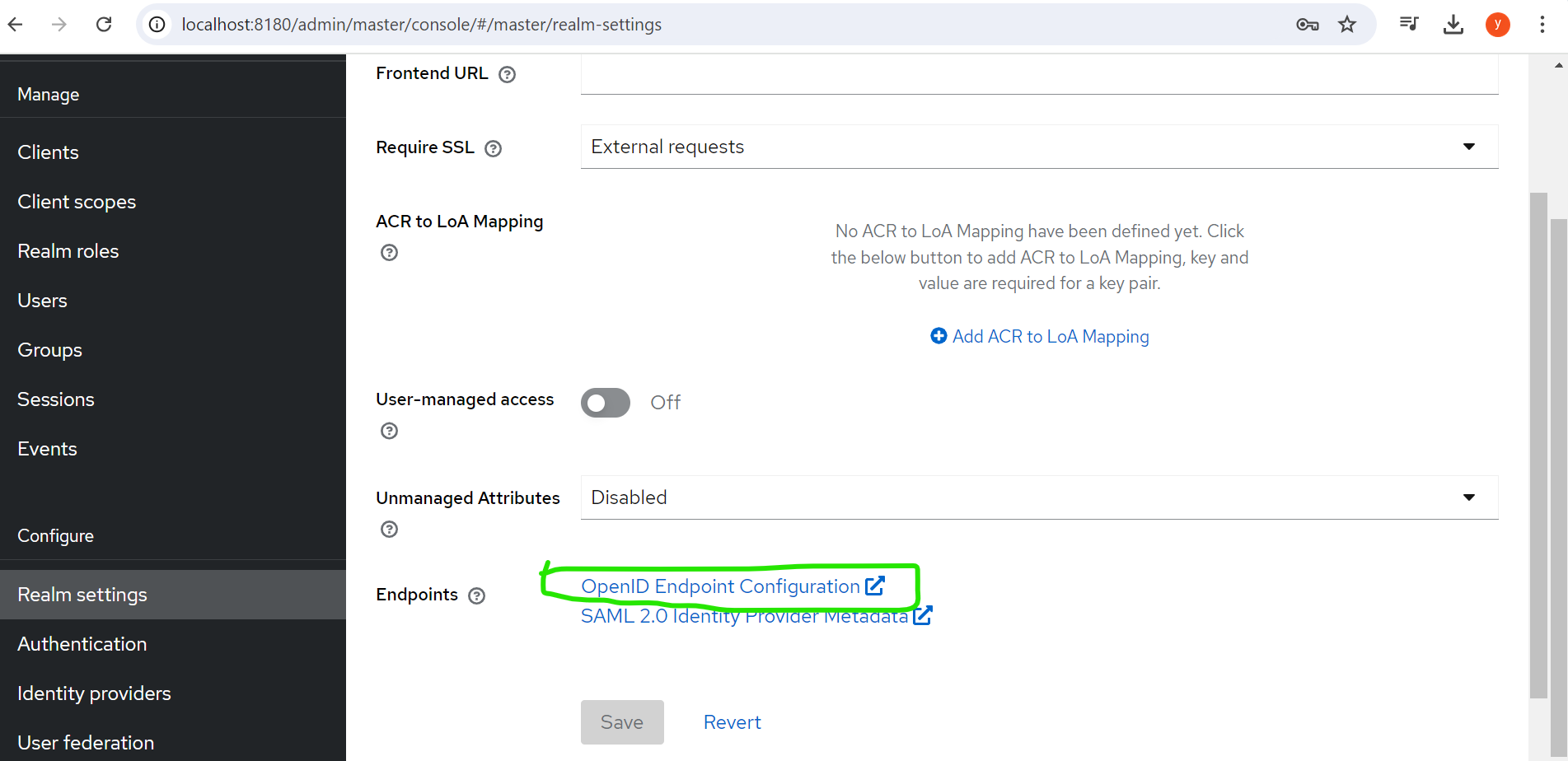
In green box is the Client\_ID (bank-call-center-cc) ,where cc is just for naming convention like Client credentials, but the Client\_ID can be anything.

Yellow box showing we opted authenticator as ClientId and Secret.

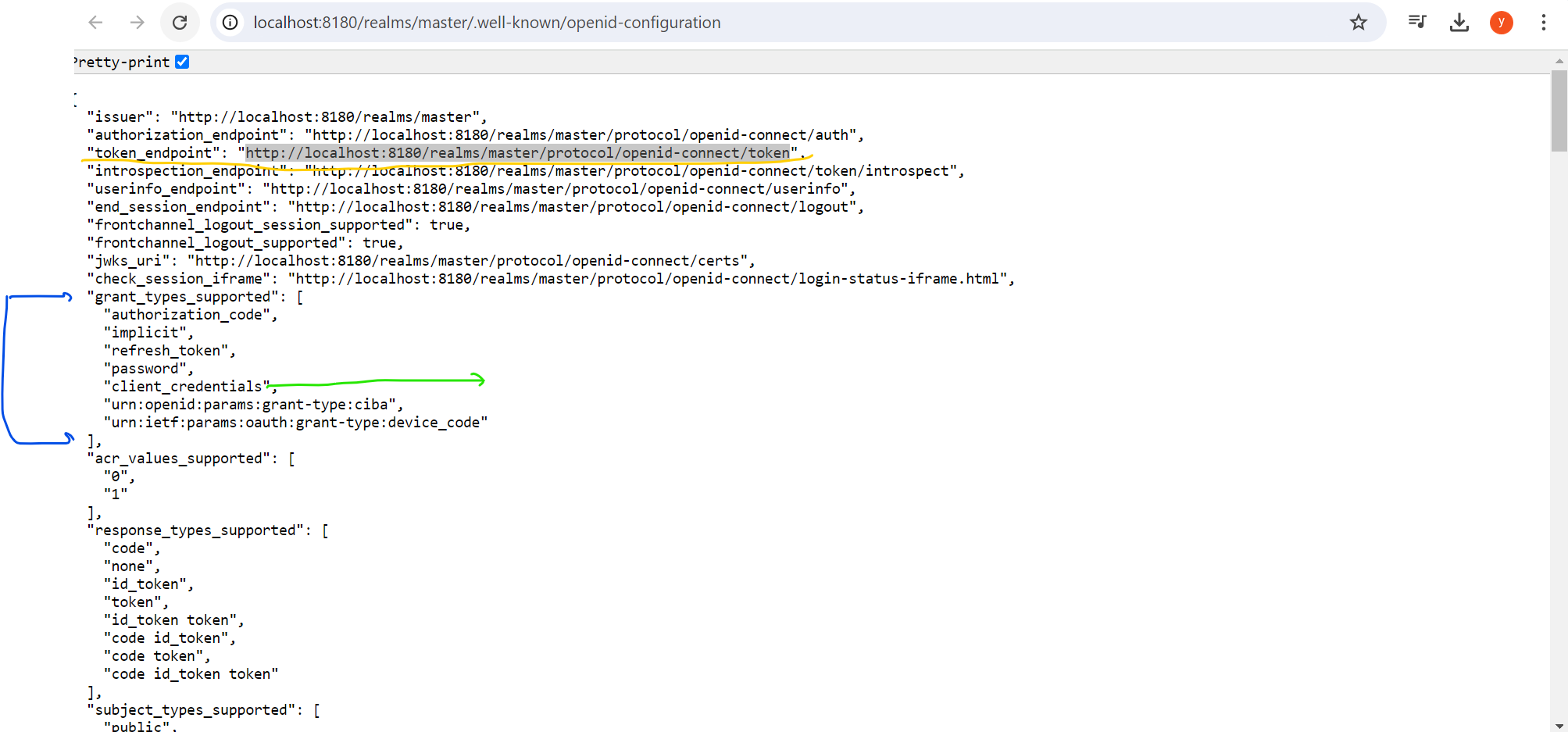
means our client need to first send the clientId and Secret to Auth server for authentication and once the authentication happens successfully then Oauth server generate the Access token .

2)Now our Client App need to sent the client credentials to Oauth server ,for this client need the

Endpoint over which client credentials will send and once authenticated the Oauth server return the access token for this

****

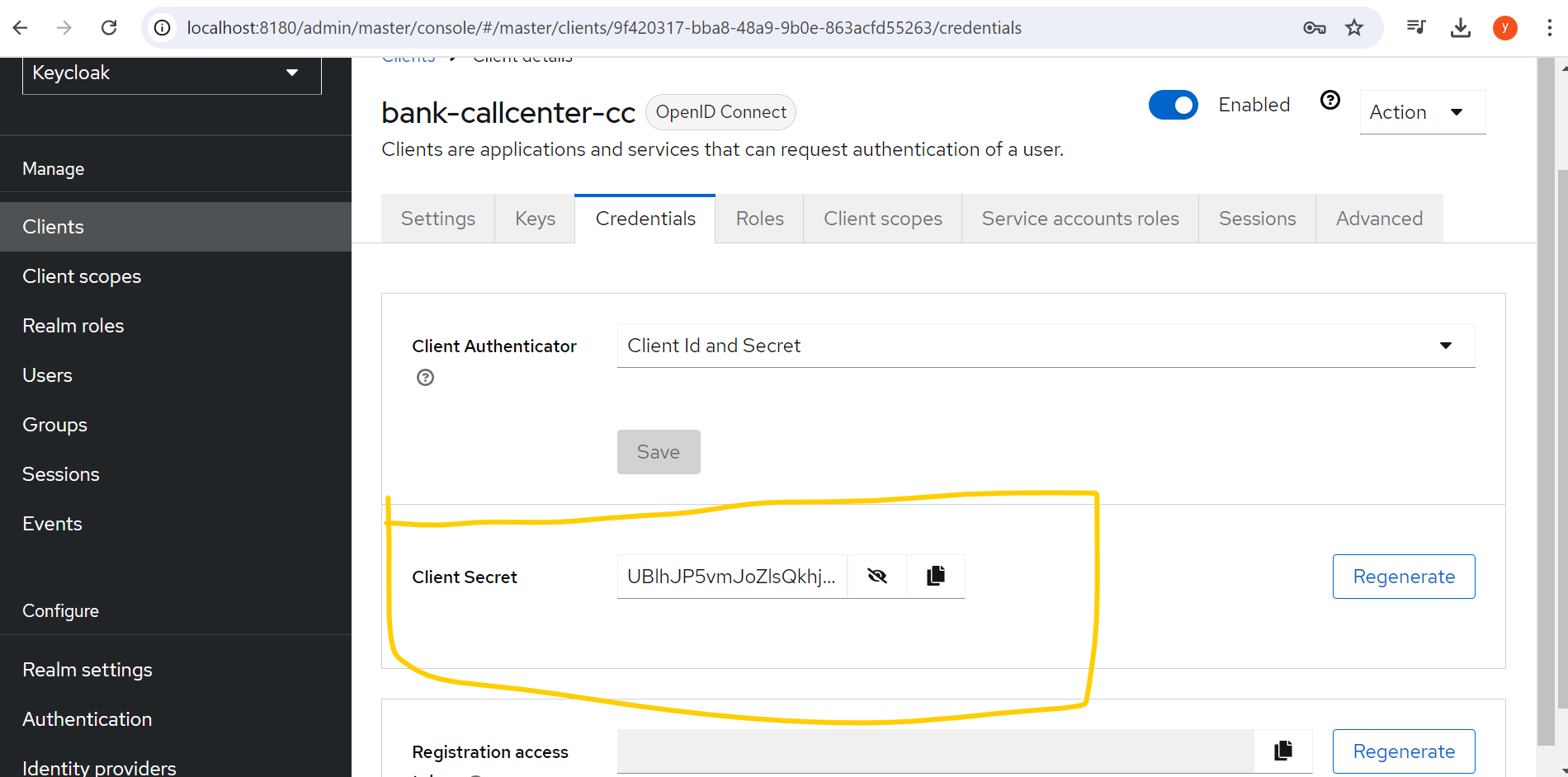
Click on this option and pick the below endpoint:



In the above picture there are multiple type of grant Types Aouth support but we are using

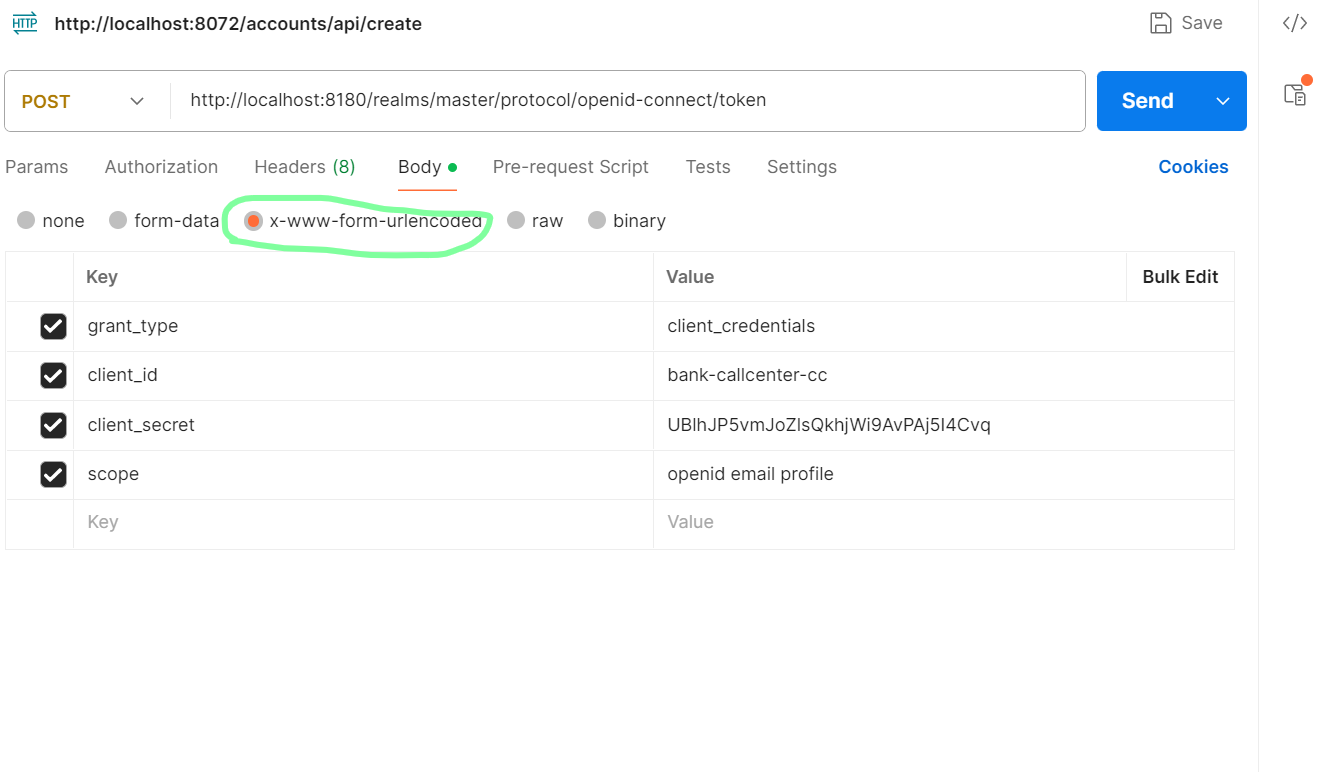
client\_credentials

Get the client\_secret as below:

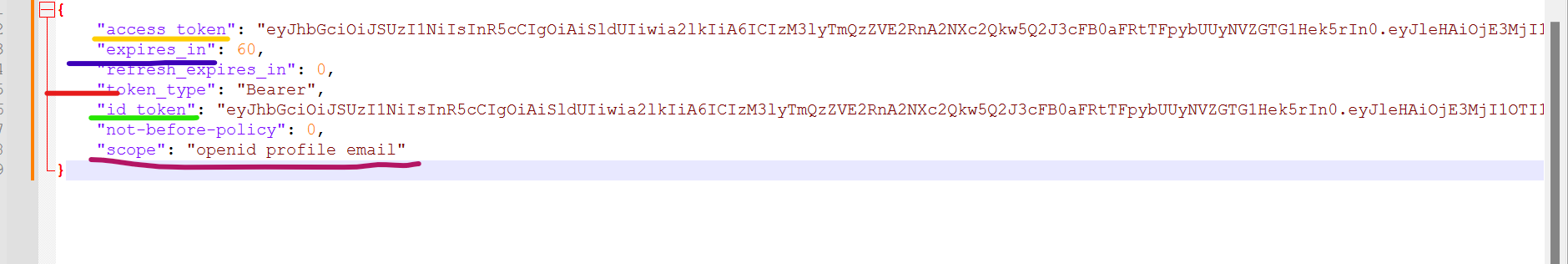


Hit the request with this below info on the above mention endpoint POST:

Note: Here the scope openid email and profile is default provided. Can check in client scope tab of key cloak



And you will get the Access token and others thing in the response Payload as json :

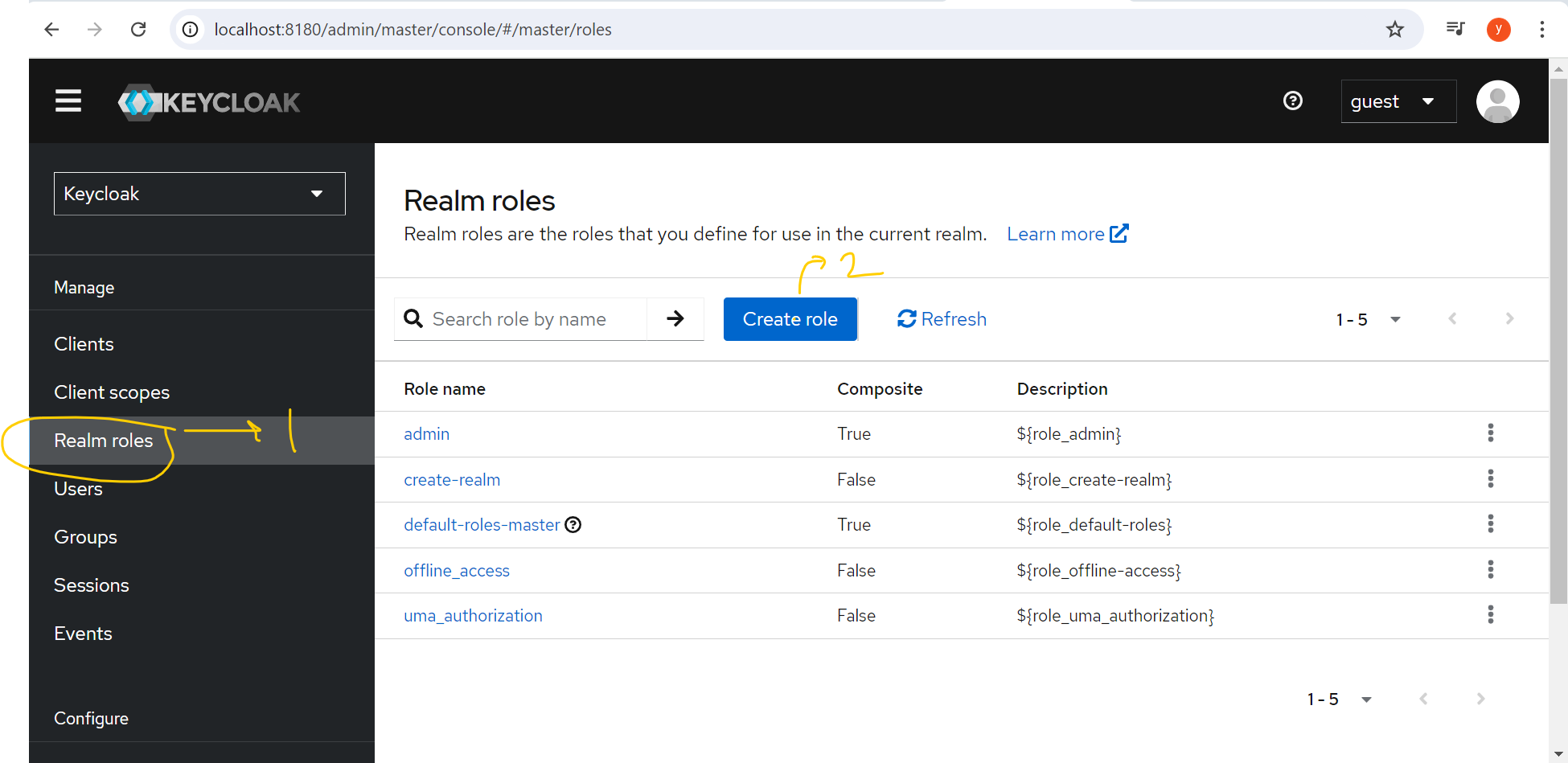


Here we are getting the id\_token because we have provided the scope openid in scope. If we remove from scope then we will not get id\_token in reponse.

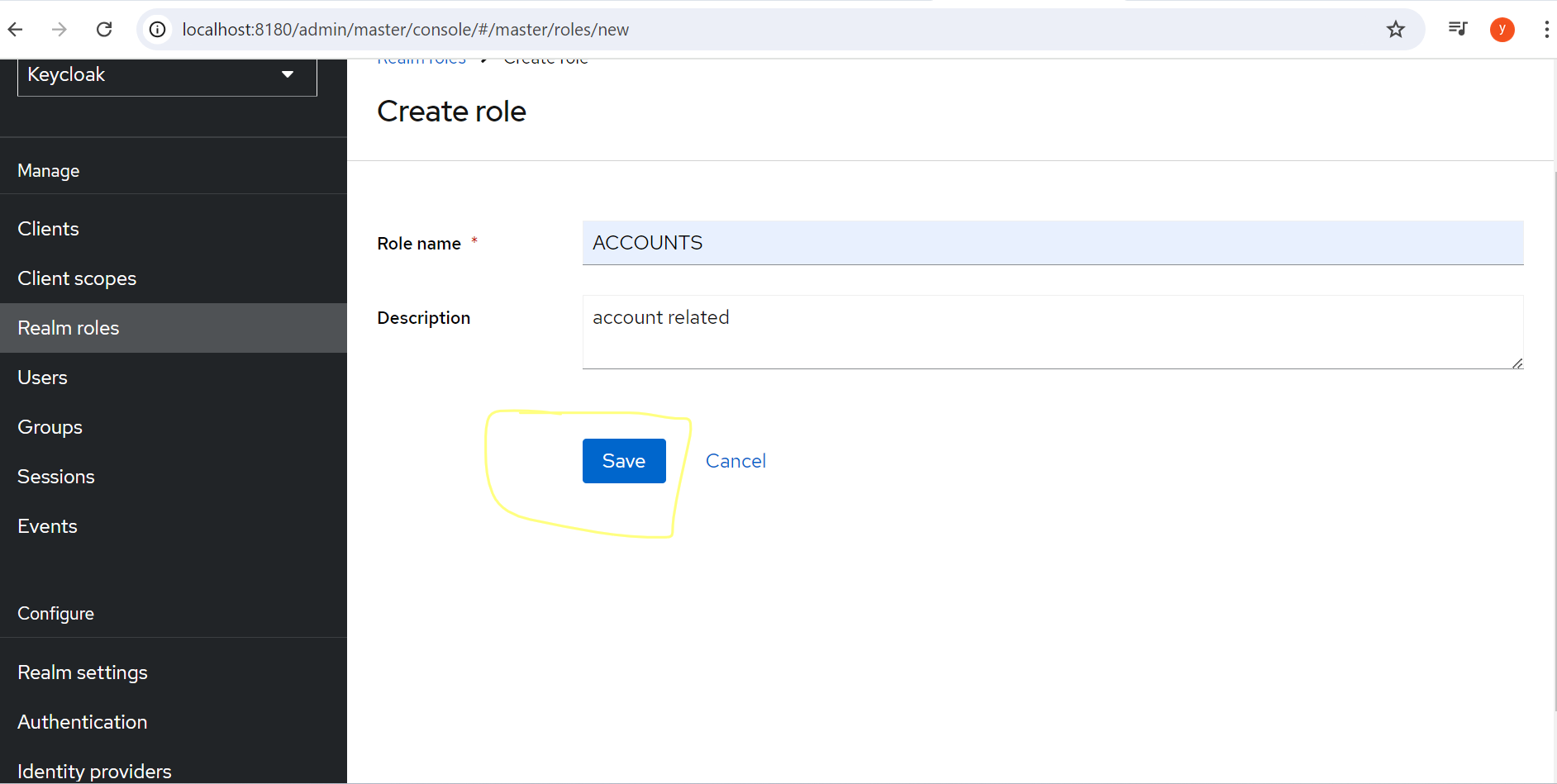
To Make the endpoint protected with ROLES:

Follow below steps:

Go to the Realm roles and click on create roles

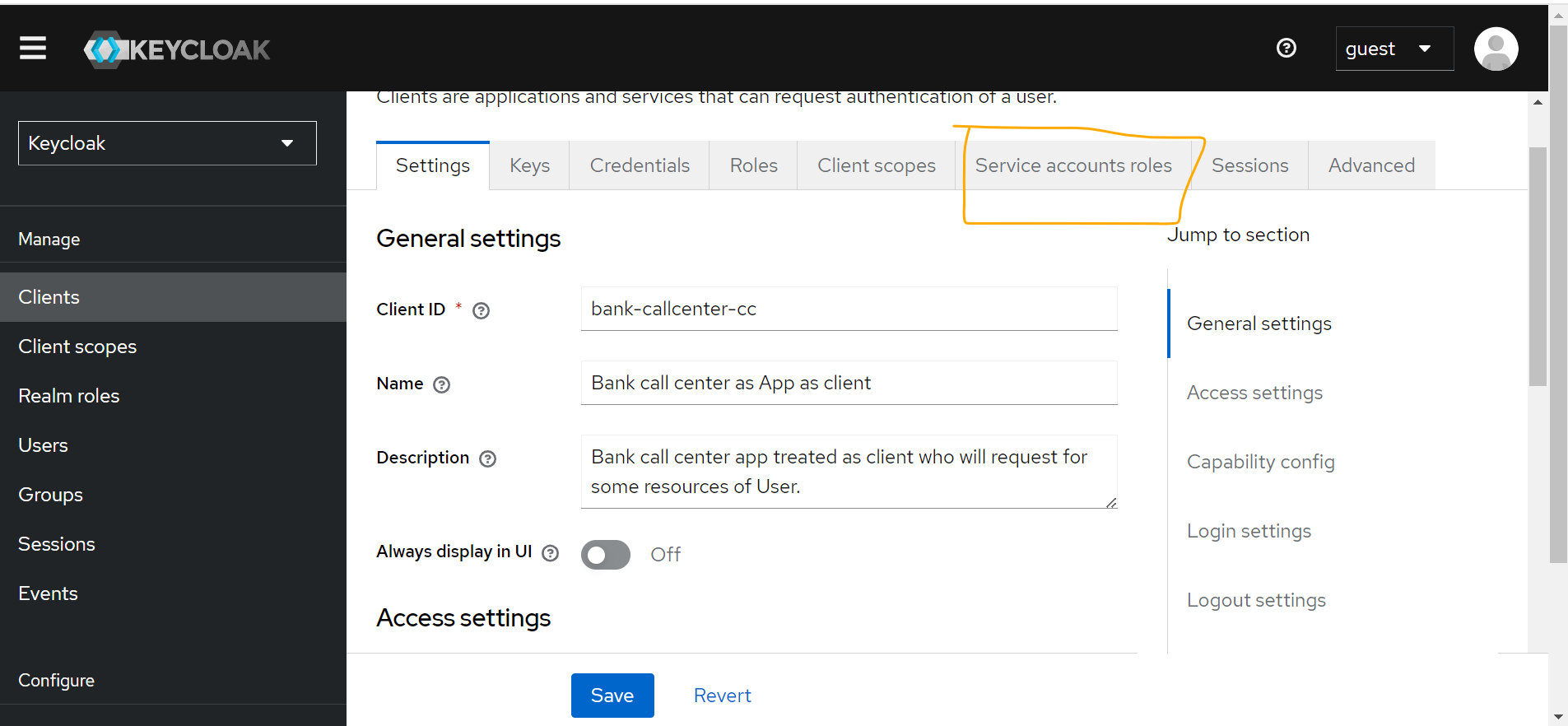


After that create the same role Name defined in GatewayServer(Resourceserver) and save.

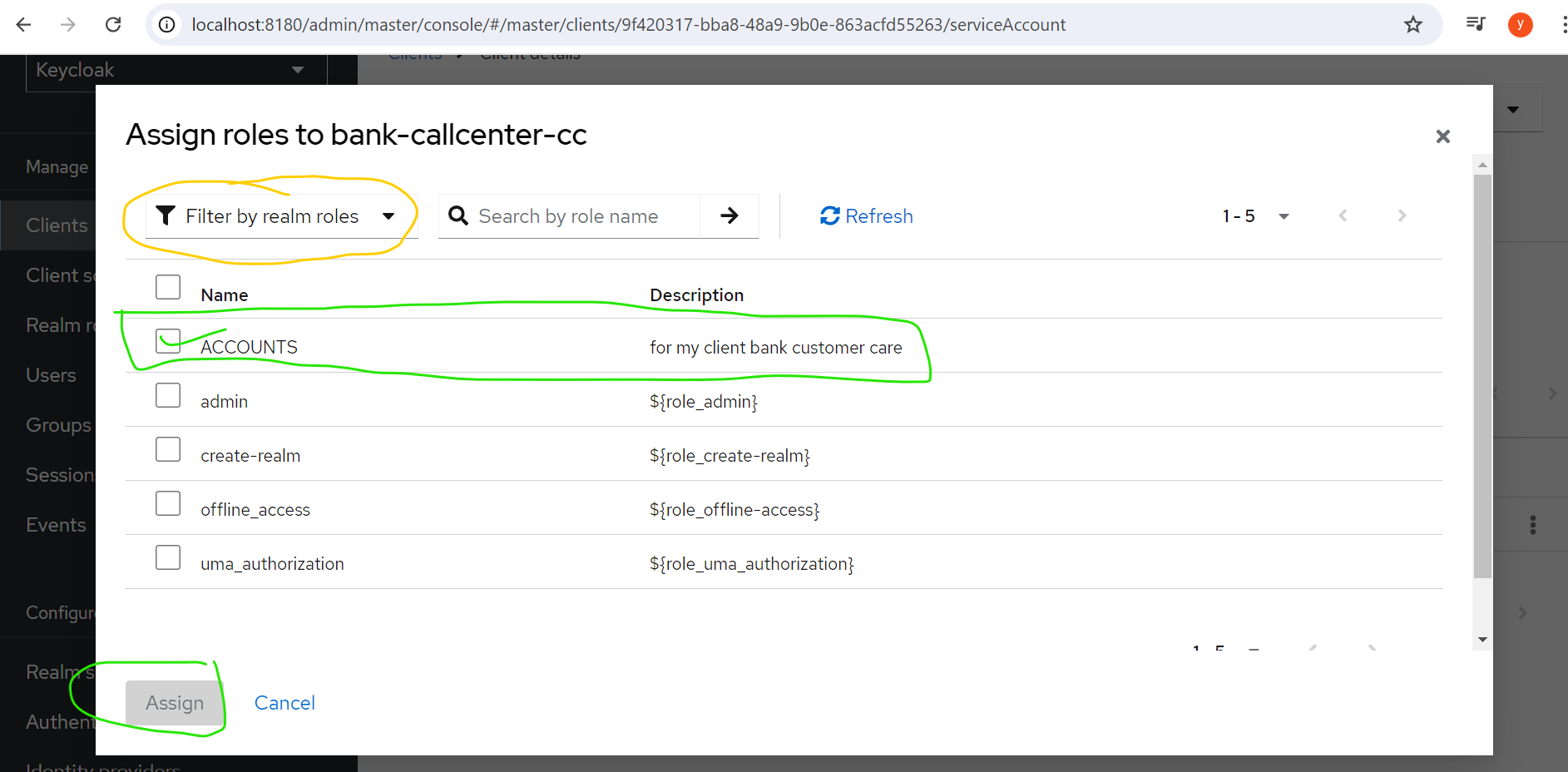


Now assign this newly created “ACCOUNTS” role to the client(bank-call-center-cc).

Under clients select the client (bank-call-center-cc) and go to **Service accounst roles** tab



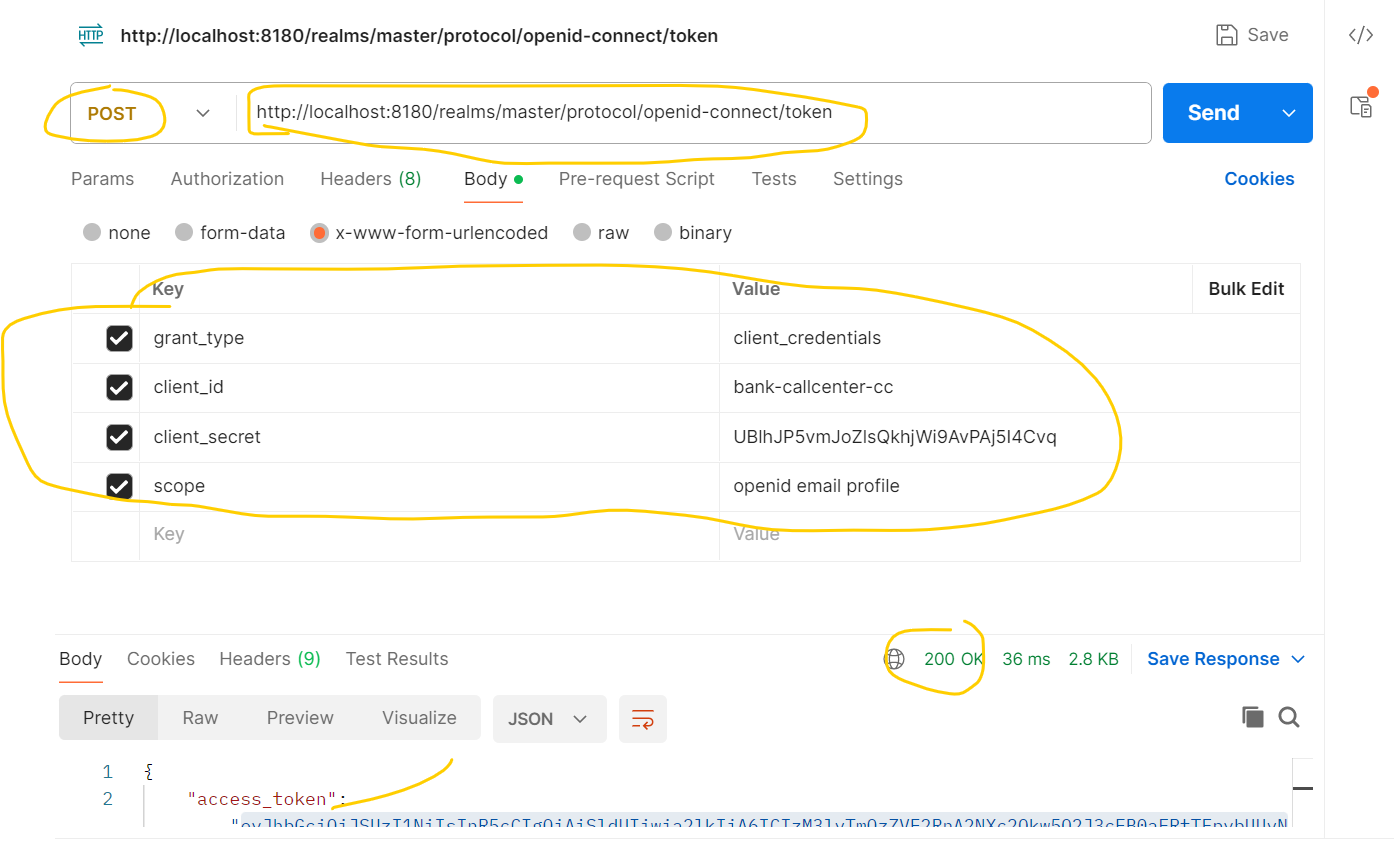
Click on assign role button select the newly create role “ACCOUNTS” as below:



Now role is assigned to client.

Now again go to the postman:

Hit the same POST endpoint :



Copy the access\_token and go to the below URL : <https://jwt.io/>

And paste the access\_Token in encode section and you can find the roles what ever you created and assign to client as below:

