

**1.Providing Permissions for users for accessing the application.**

Instead of allowing all users in realm to access the application, instead we can provide access to specific users in a realm to particular routes.

For that we have to create required resources, policies and permissions in authorization tab in client details.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Among last 4 images , first image shows the root page of the application and it has two routes “http:publicip:7000/sea” and “http:publicip:7000/mountain”

Remaining 3 images shows the resources , policies and permissions requires for accessing “root” and “sea” and “mountain” routes.

**a)Resources**

Its nothing but just the path or routes which we want to secure, we should create it as a resource.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The above 3 Images shows the routes to be protected “/”, “/sea”, “/mountain”, which will be created as resources.

**b)Policies**

A screenshot of a computer

Description automatically generated

Here we are giving access to the resources using users based policies but we can give access to users using roles based policies also. Above are the list of users.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The policy named “Root” includes users user-b (19995def-…) and user-g (7452dffa-…) , Similarly policies name “Sea” includes user “user-b (19995def-…)” and “Mountain” includes user “user-g (7452dffa-…)” only.

**C)Permissions**

The Permissions in Client Authorization tab is used to give permission to the user for the client routes by linking the resources and the policies already created.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The above two Permission are resource based permissions which links resources and policies to provides access to the resources according to policies.

First one shows linking Resource named “root” and policy named “root”, similarly the second images shows linking Resource named “sea” and policy named “sea”.

A screenshot of a computer

Description automatically generated

The above images shows linking Resource named “mountain” and policy named “mountain”.

By Linking these resources and policies “user-b” has provided access to <http://publicip:7000/> and <http://publicip:7000/sea> and “user-g” has provided access to <http://publicip:7000/> and <http://publicip:7000/mountain> , where <http://publicip:7000/sea> and <http://publicip:7000/mountain> serves an image of sea and mountain respectively.

A screenshot of a login screen

Description automatically generated



When we enter <http://publicip:7000> it redirects to keycloak login UI, when we enter the credentials of user-b , it gets into root page of application.

A screenshot of a computer

Description automatically generated

A close-up of a wave

Description automatically generatedThen we get the following results if we click sea and mountain for user-b.

A screenshot of a computer

Description automatically generated

Similarly if we enter “user-g” credentials ,we get the following results

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Because while configuring in keycloak we have given “user-b” and “user-g” to access only sea and mountain routes respectively while restricting access to other route.

**NOTE:**

Keycloak doesn’t support RBAC or any other user based access restriction natively, we have to modify the application code to match the configuration. Here in this case , I have provided resources name (root, sea, mountain) in the application code which is the same I used in the keycloak configuration.