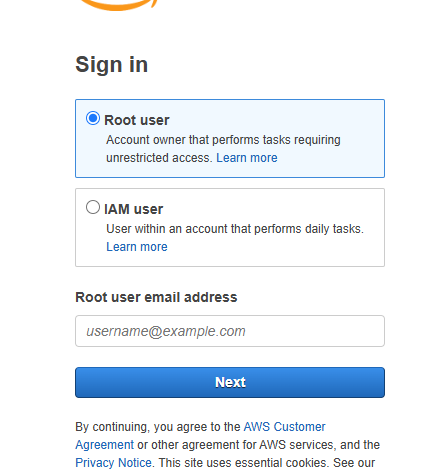
AWS IAM (Identity & Access Management)

We are able to log into AWS cloud and use services -> two types of accounts -->

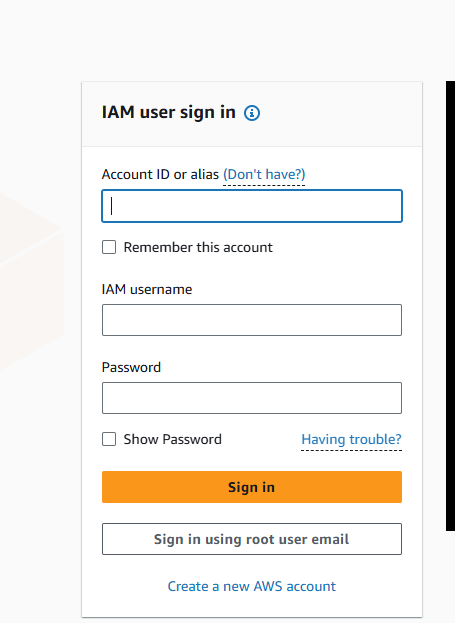
Root account and IAM account

Root is like the super user and no limitations. It is the most powerful account with no restrictions

If you want to place restrictions on user accounts, then we need IAM user. I can decide which user should have what access



Sign into AWS console, that’s what we usually see



It is used to manage users, groups, policies and roles. IAM is a free service

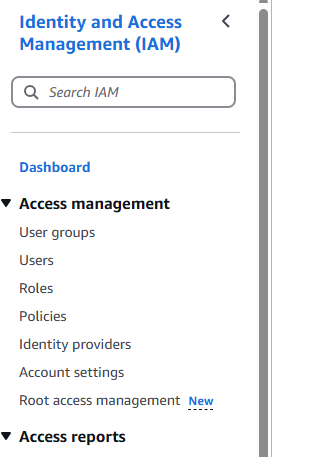
Root account is very powerful with no restrictions and we can access everything in AWS cloud

Key points to be considered for the root account:

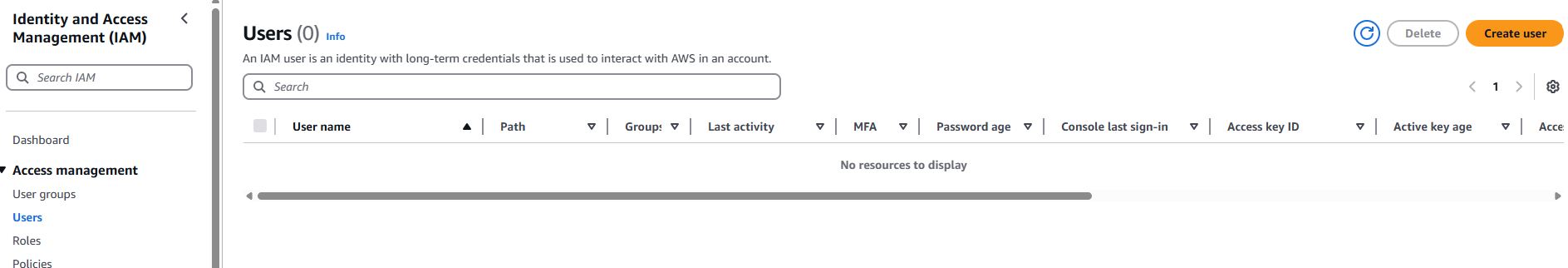
We should not use root account for daily task also we should not share root account credentials with anyone (it is also highly-recommended to enable high security by enabling Multi-Factor Authentication MFA for root users)

As a part of a project or team, we will not be getting root account credentials rather we will get IAM account credentials with specific access

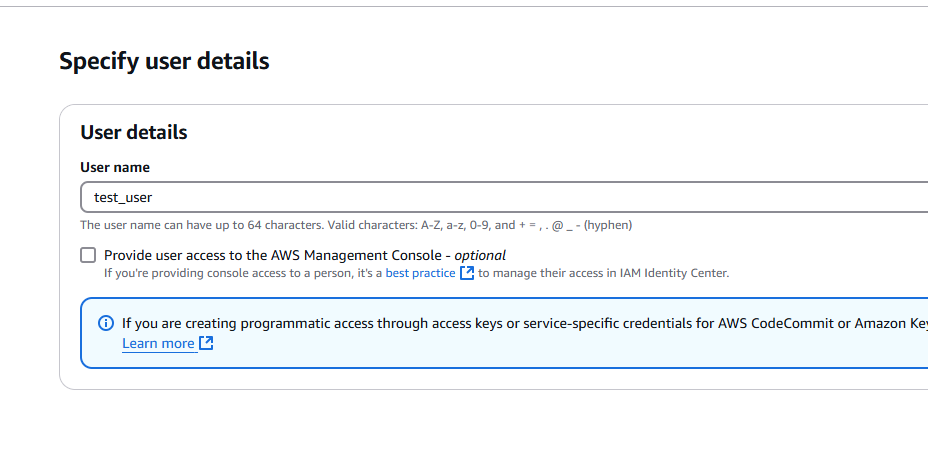
Open IAM



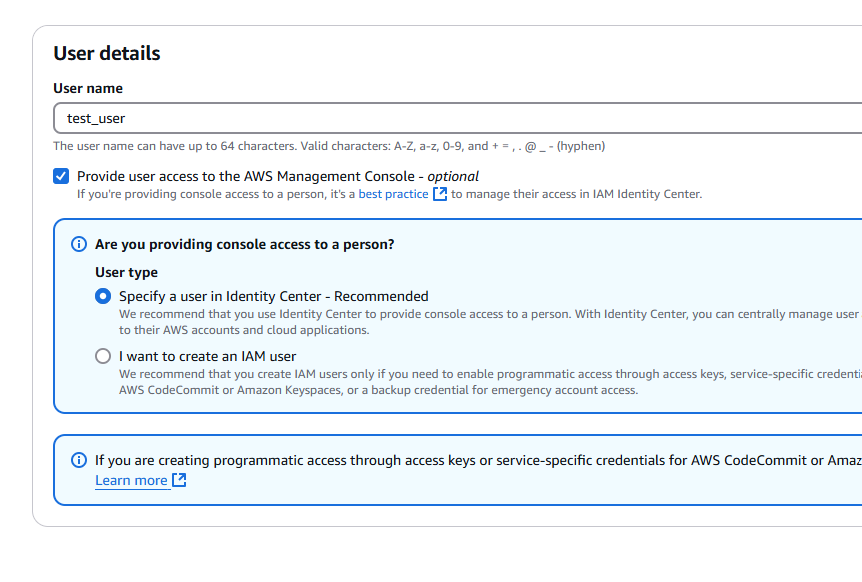
Click Users



Create user

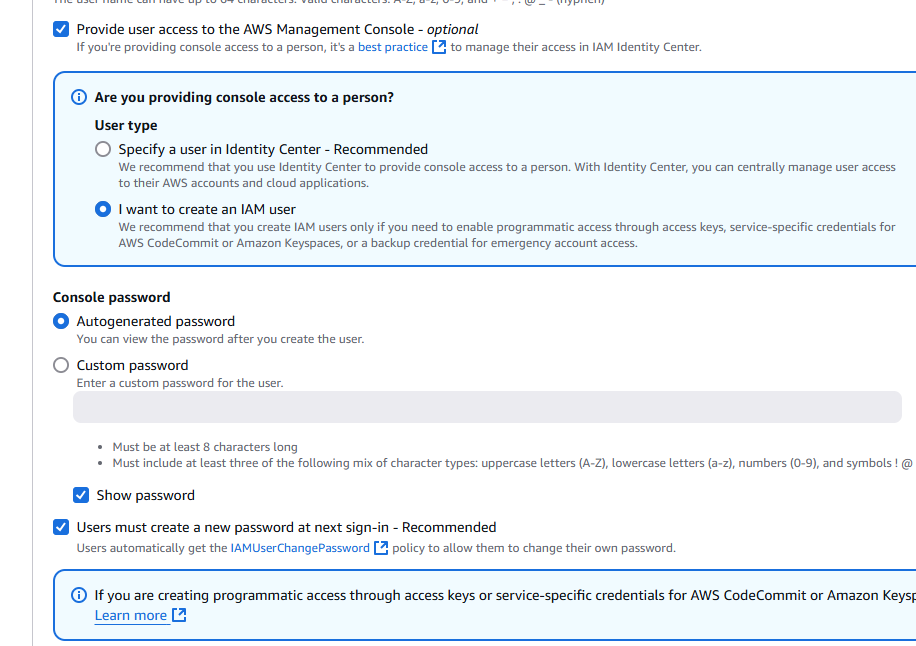


If you don’t click ‘Provide user access to the AWS management console - optional’ then user cannot access AWS console through browser

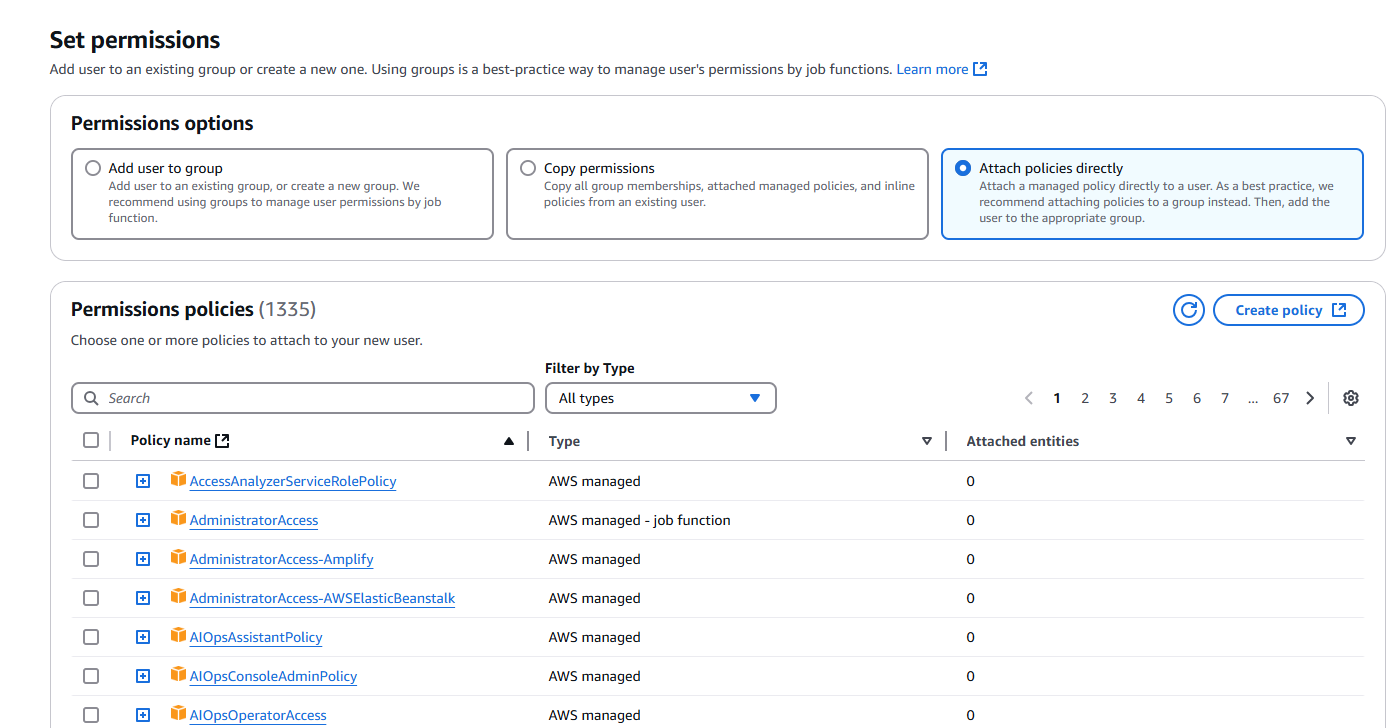


Apart from browser access, AWS console can be accessed via AWS CLI, Terraform, Software Development Kit (SDK). Using Security key or Access key only we can access AWS console

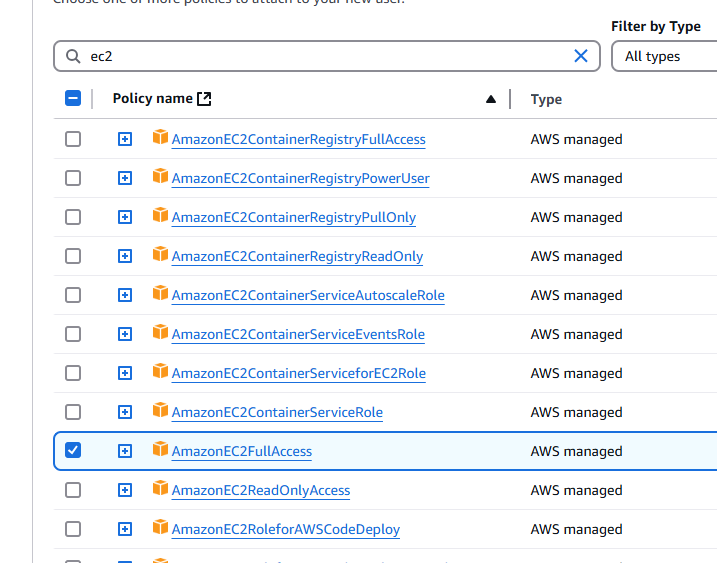
Lets use the second option



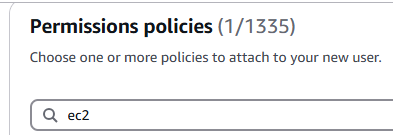
‘Attach policies directly’ means what the user can access

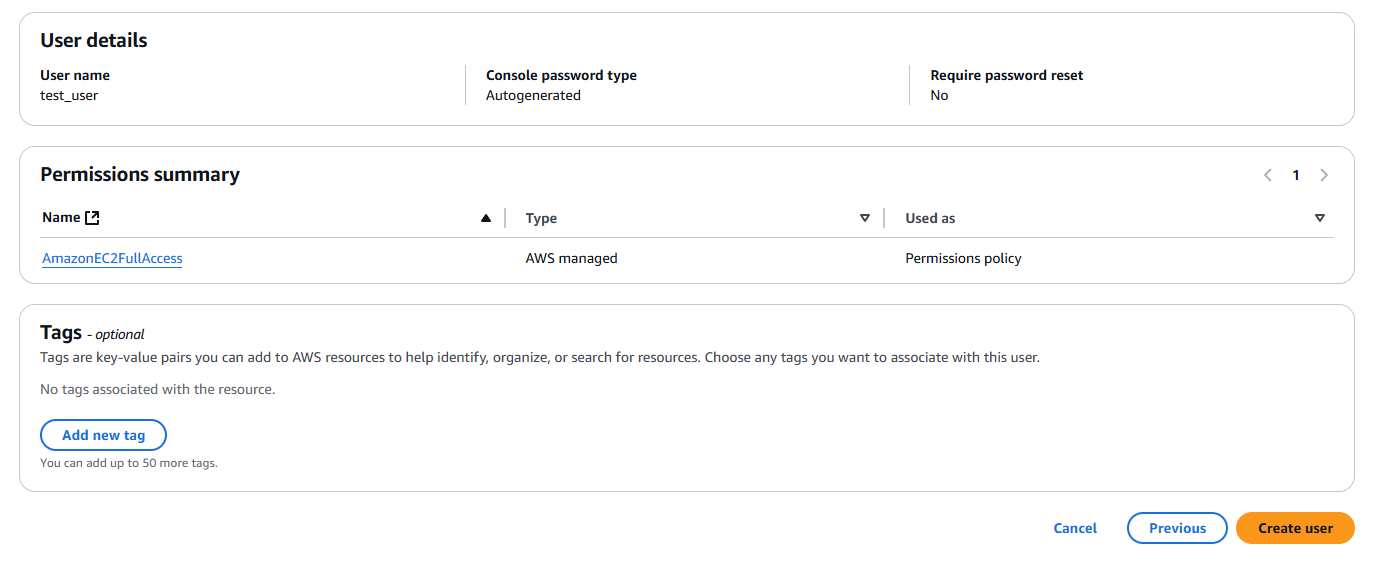


I want to give only EC2 access



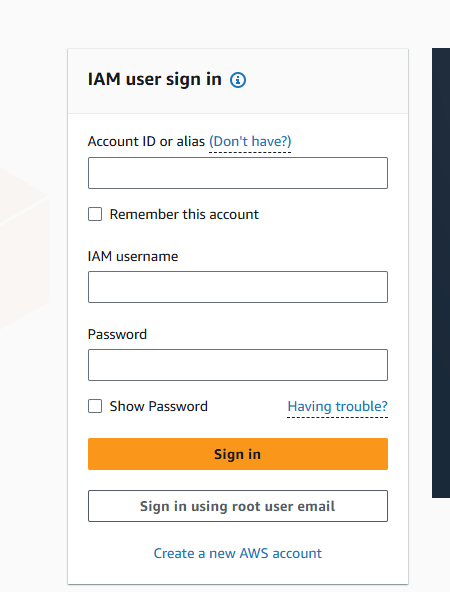
In total, we can give 1335 permission policies to your user ---> AWS managed policies

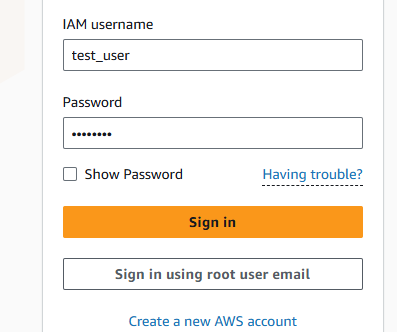




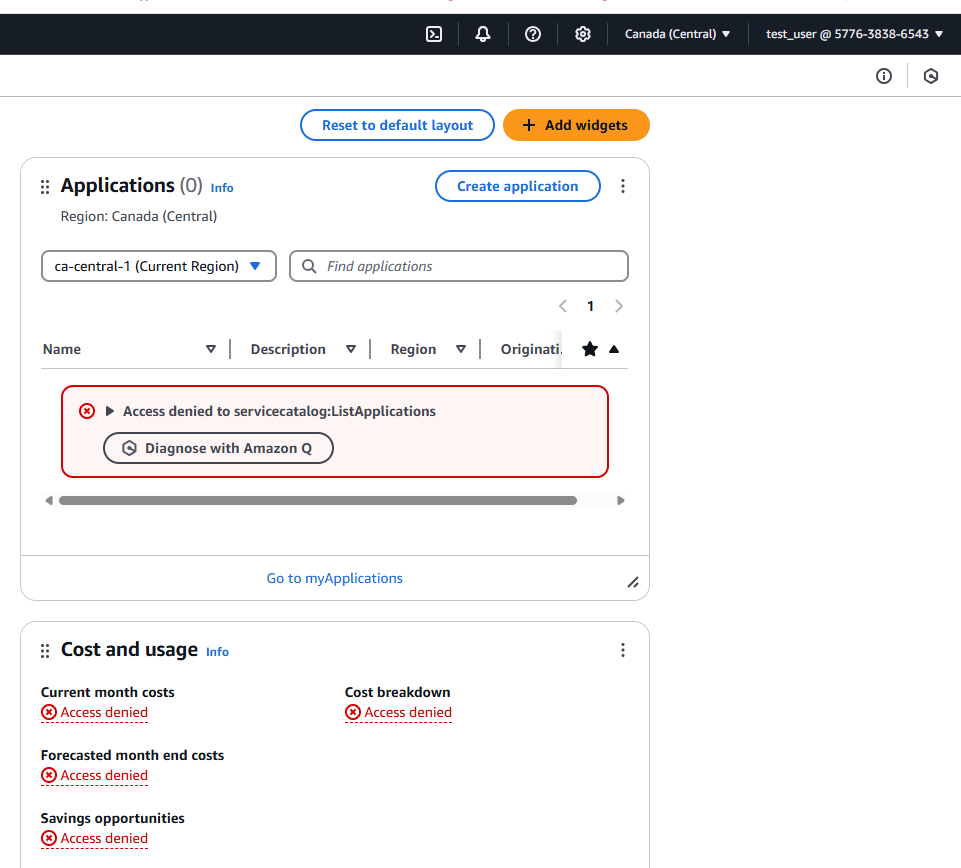
Click Create user

Sign out of account



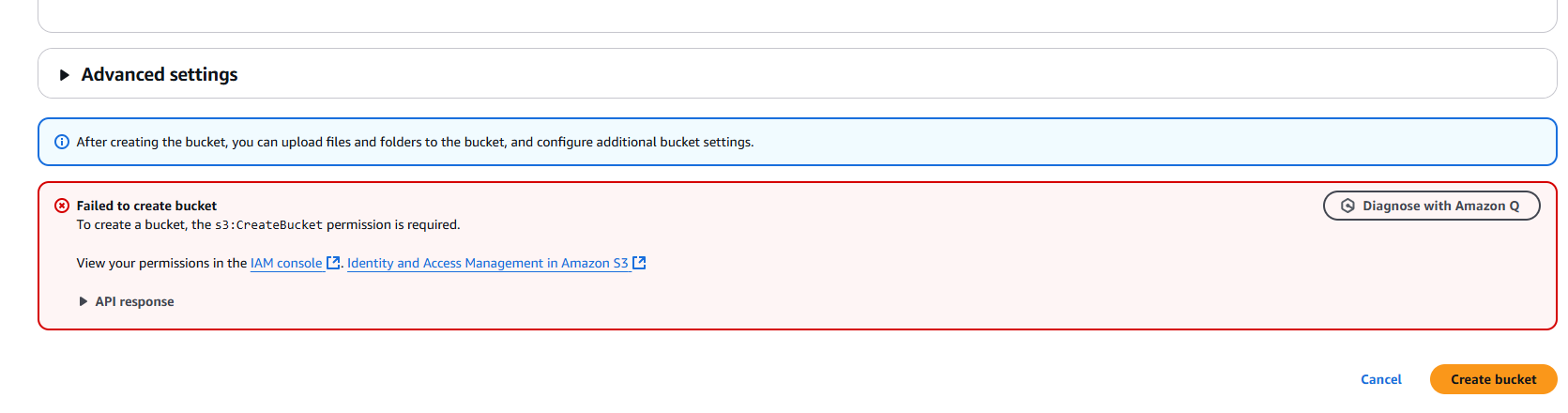


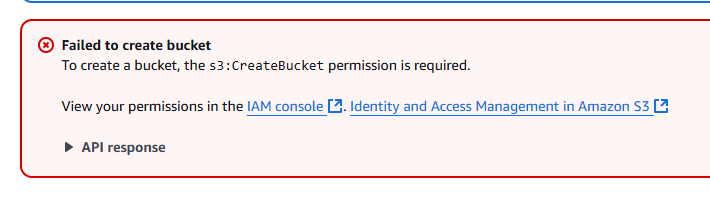
For example, Cost and usage is access denied



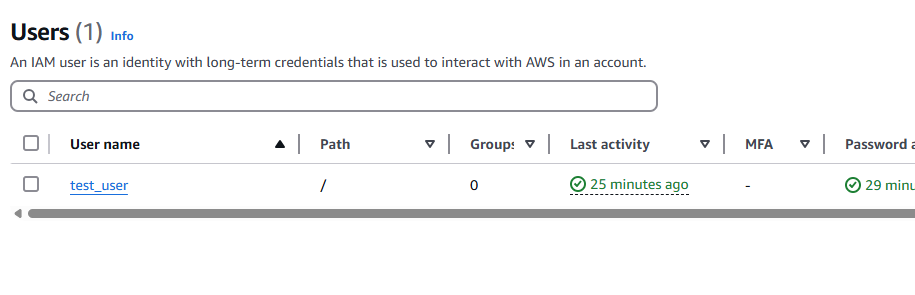
Only EC2 access is there for this user

Access denied when trying to create S3 bucket with the test\_user

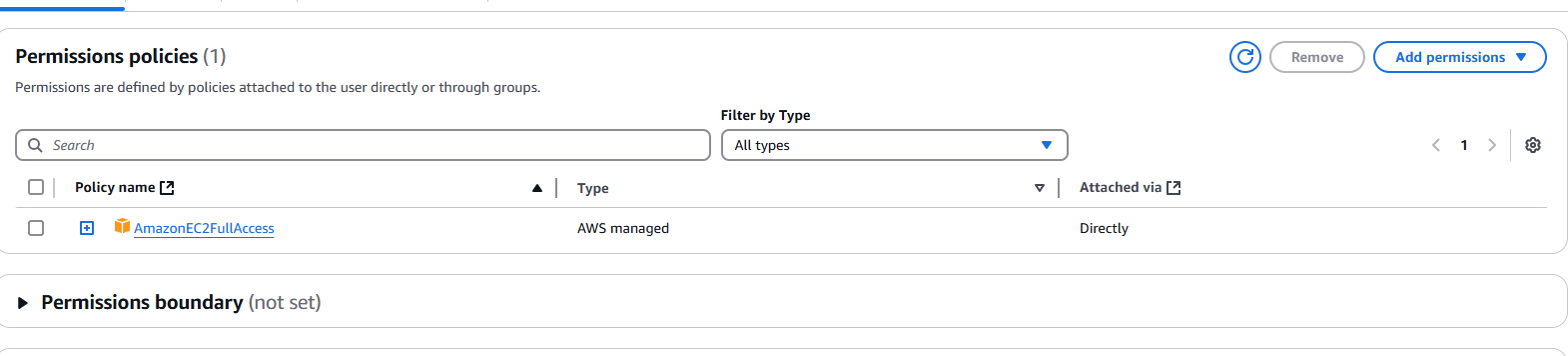




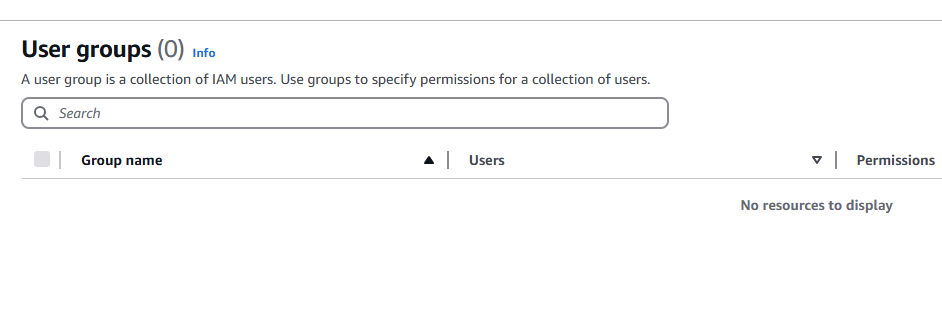
Login as root user



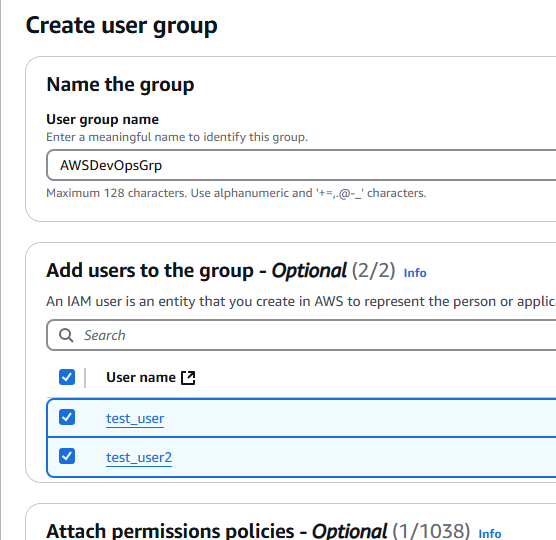
Later also I can modify Permissions policies



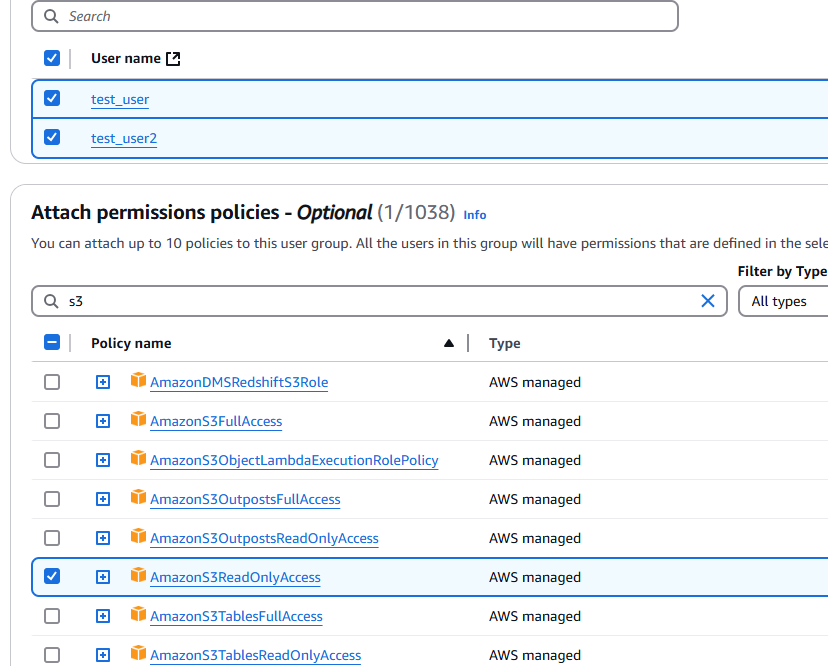
User groups are also there



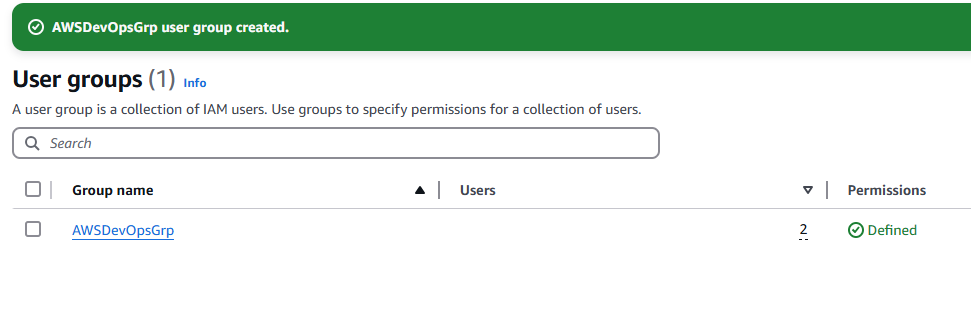
I add users to user group



I add permissions to the user group

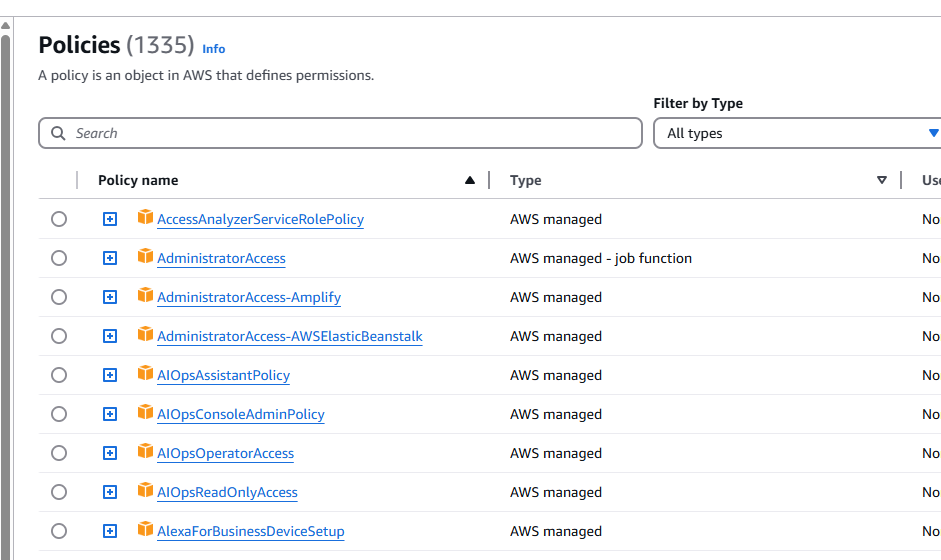


User group is created



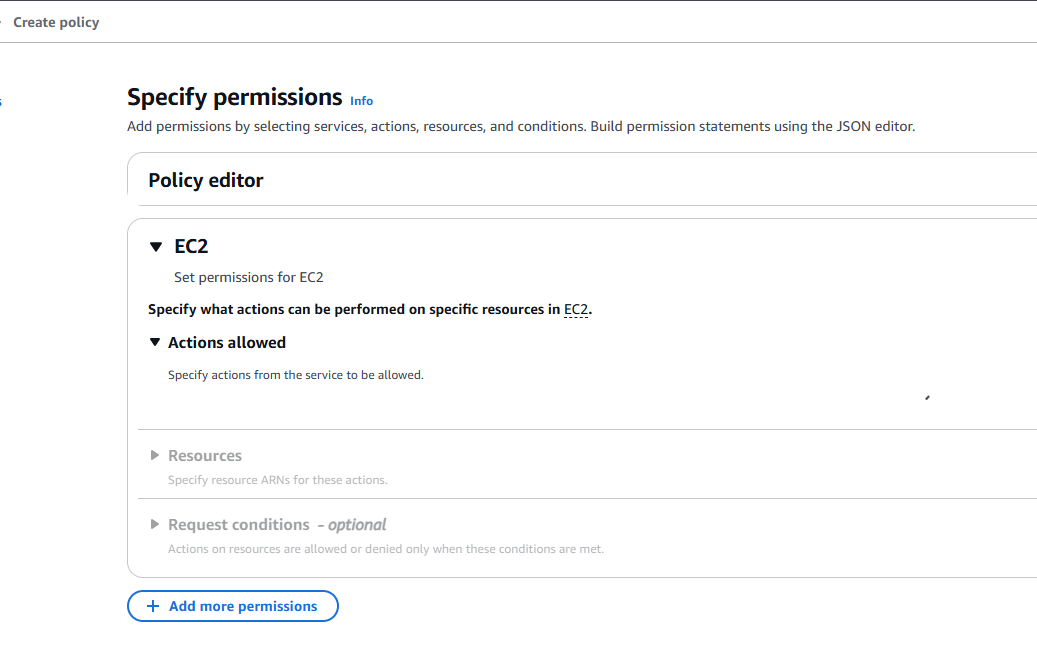
Policies

AWS managed policies

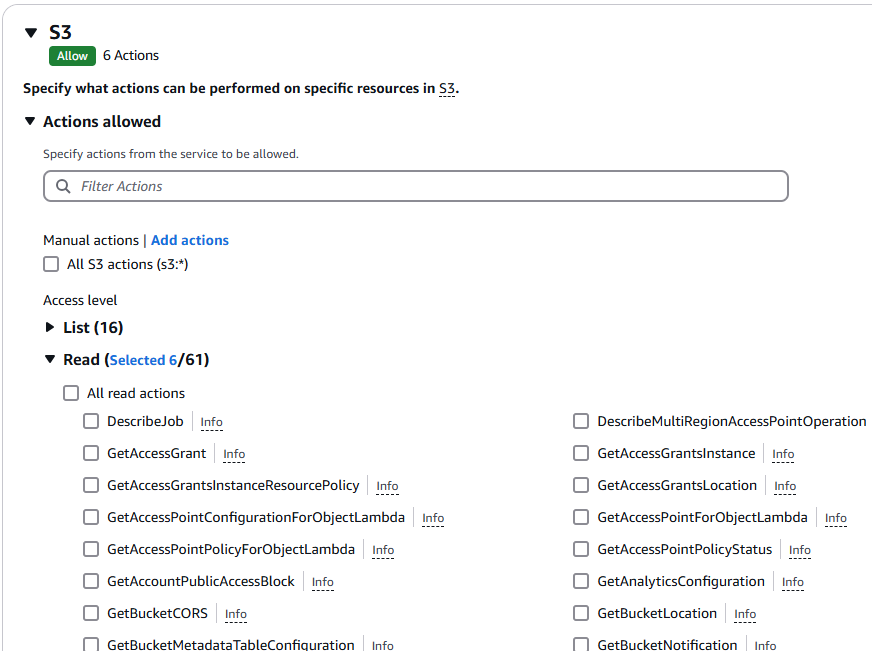


How to create my own policy?

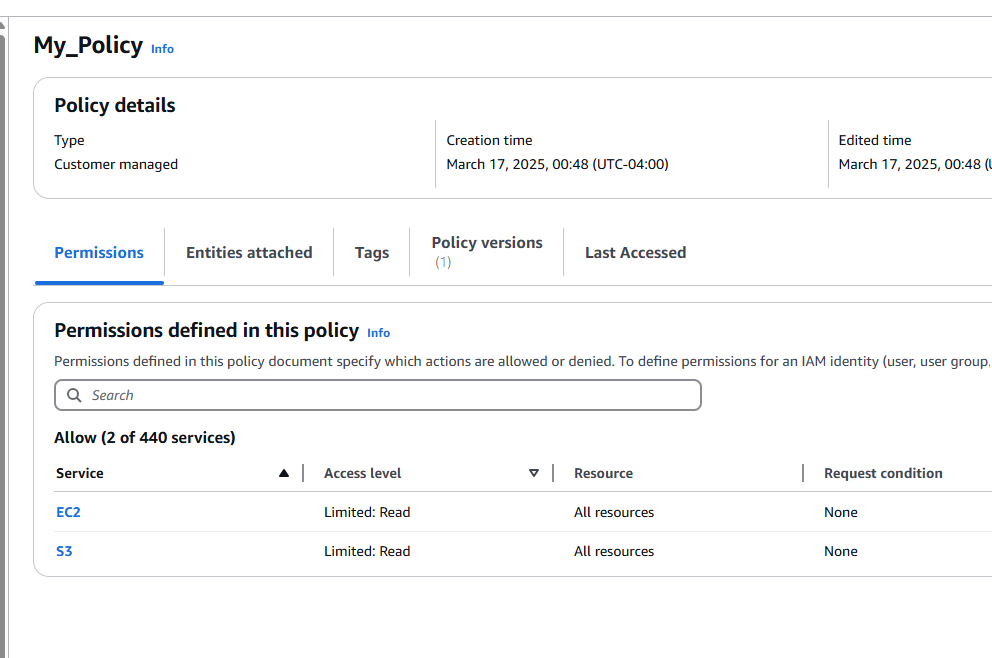
Click Create policy



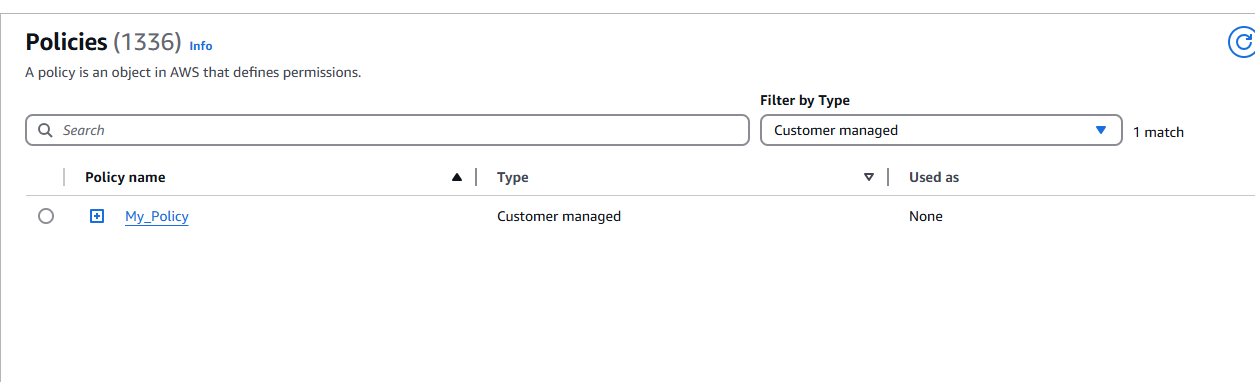
I can select whatever policies I want to include



I have created the custom policy

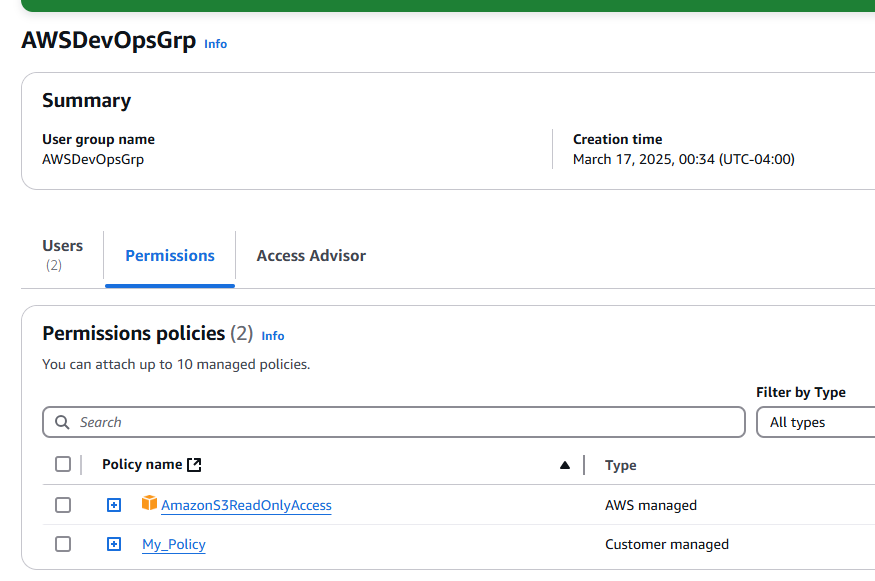


I have got an extra option here: Custom managed

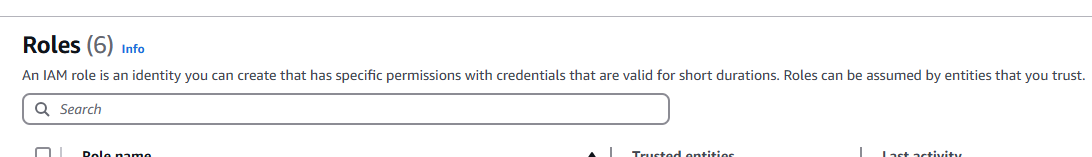


We have two policies for this user group

One is AWS managed and other is Customer managed



Roles



IAM Roles: Inside AWS cloud we have a service EC2, say if the root user has given you (IAM User) a policy to access EC2. can you access EC2? Yes. From EC2, IAM user is accessing EKS, Beanstack then you need to specify Role

EC2

Policies Permissions is sufficient here

We need Role here

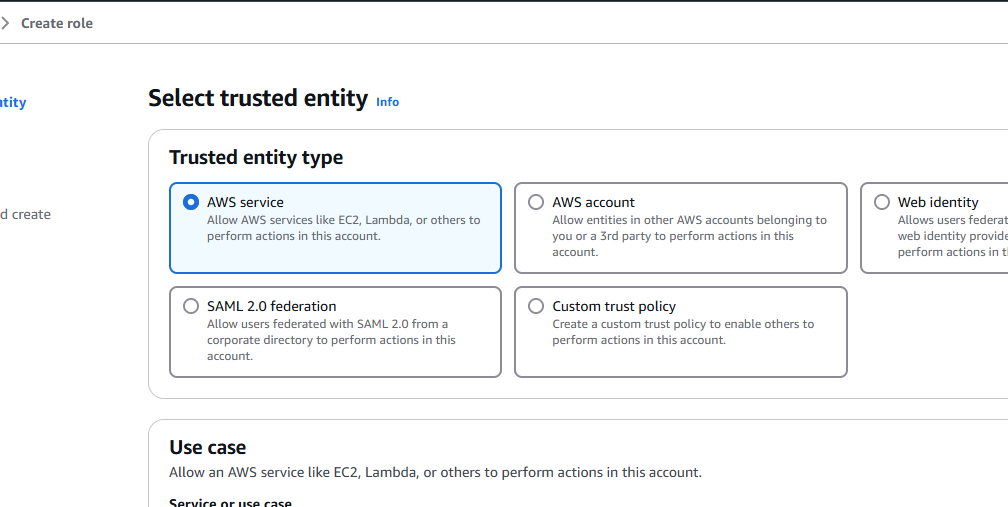
EKS cluster

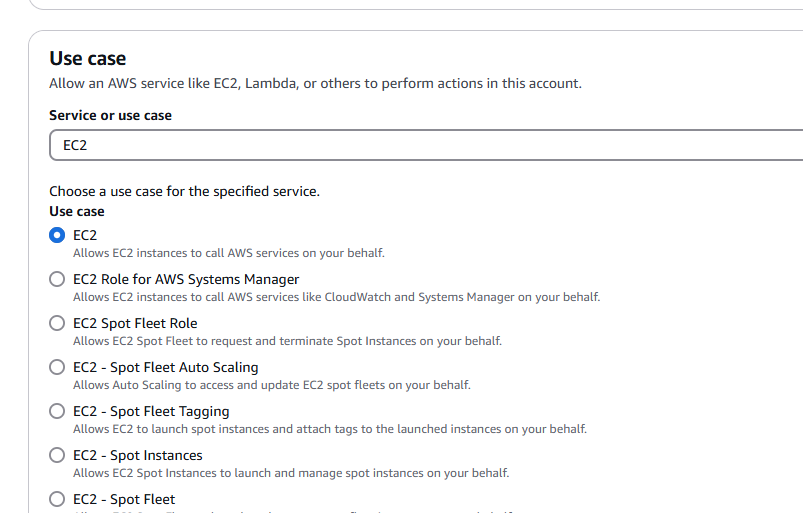
IAM User

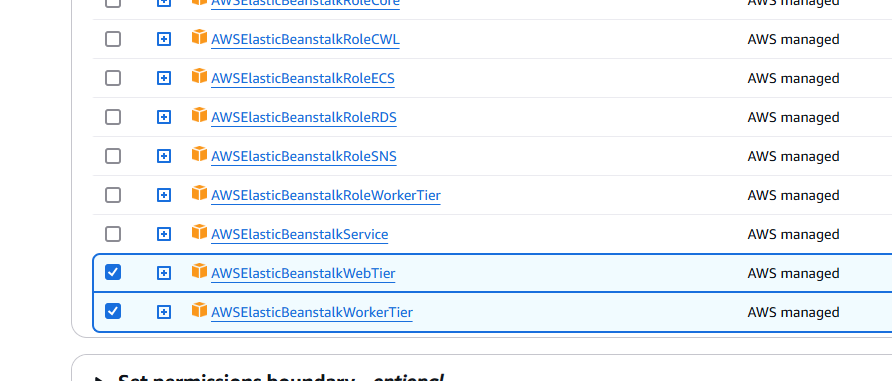
Beanstalk

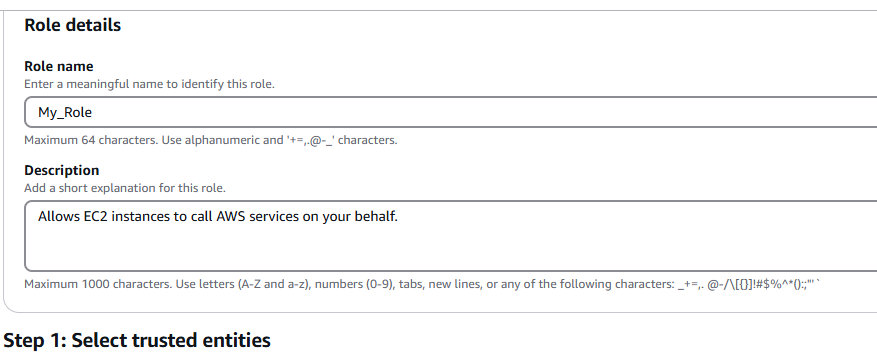
We (IAM users) need ‘Role’ to access one service from another service.

Which Role you want to create?



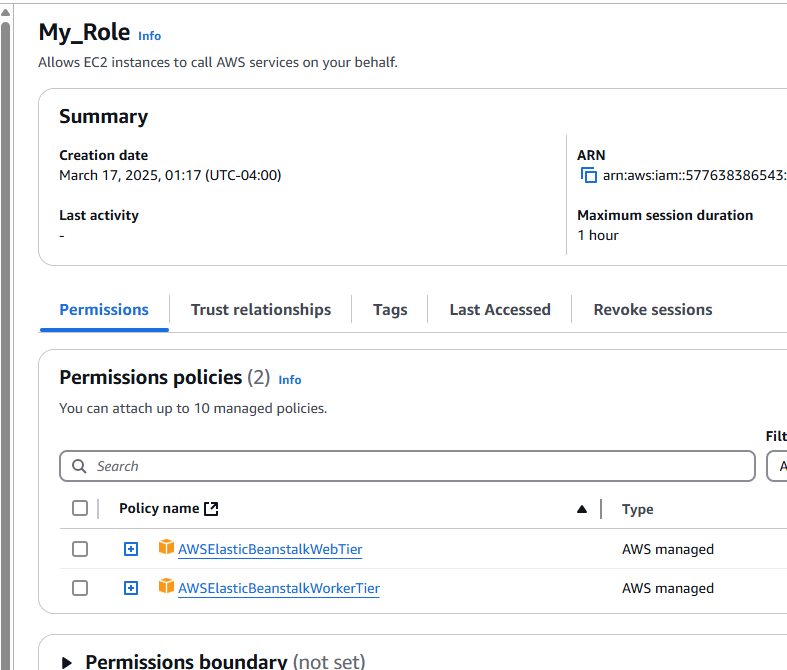




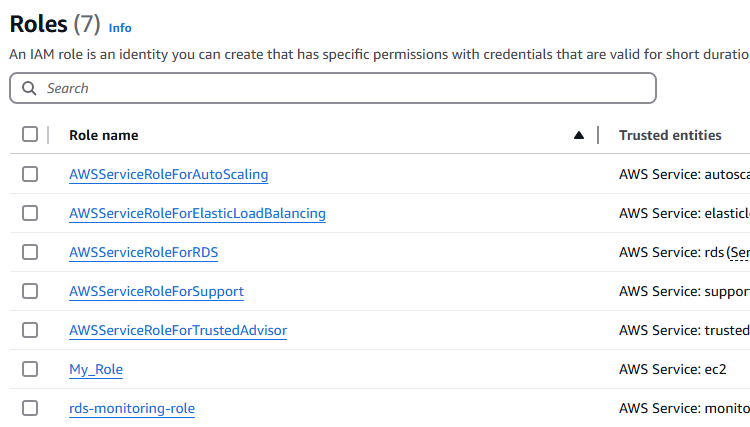


Allows EC2 instances to call AWS services on your behalf

Click Create Role

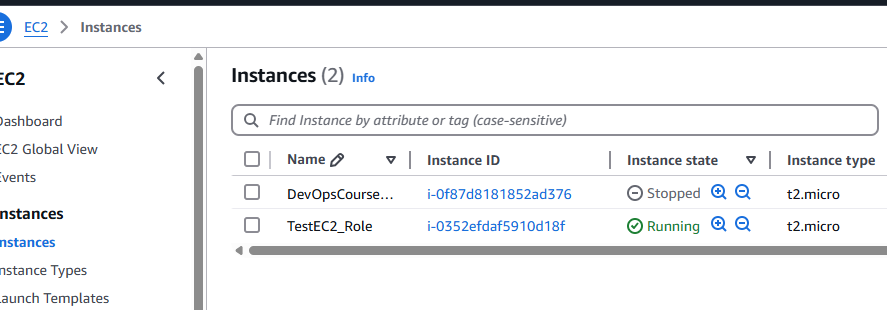


My\_Role is added to the list of Roles

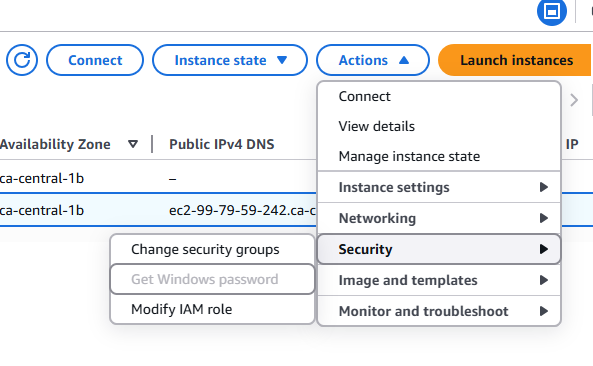


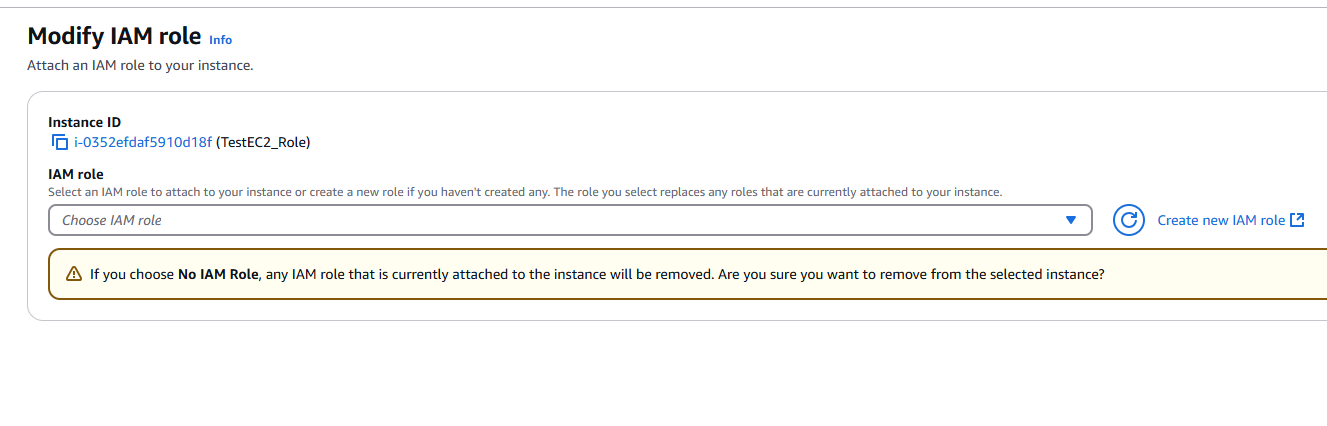
Role demo:

Now I got into EC2 and create an instance

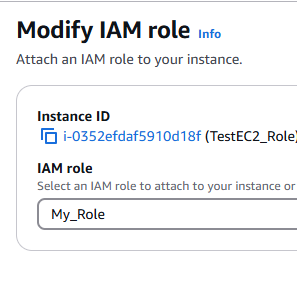


Click on Actions ---> Security ---> Modify IAM role





I select My\_Role



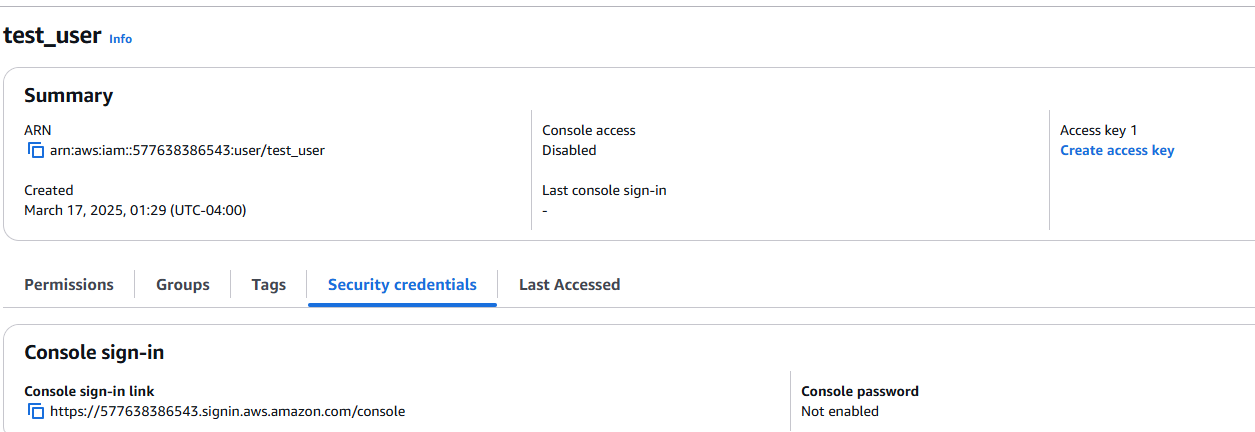
Click Modify IAM Role

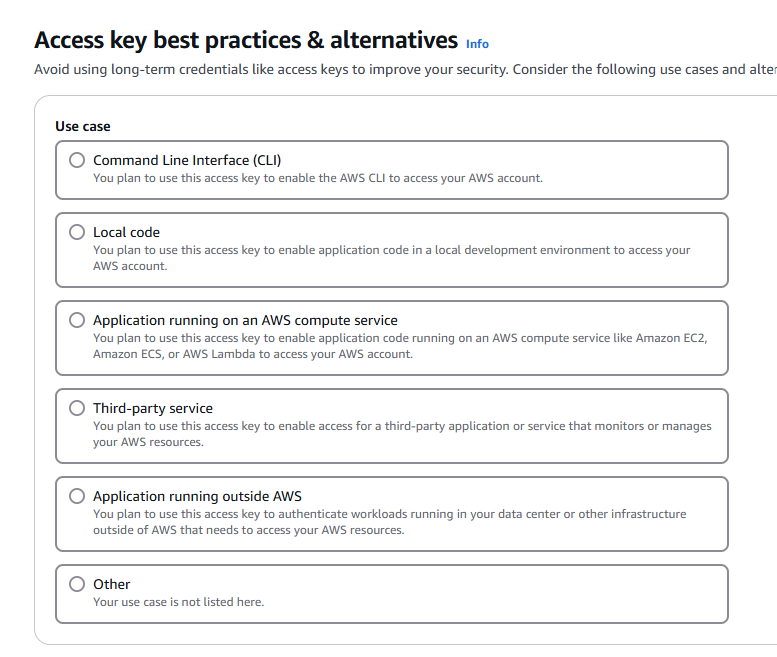
**Now can you access Elastic BeanStalk from this EC2? YES**

Role provide Service to Service access

I create another User

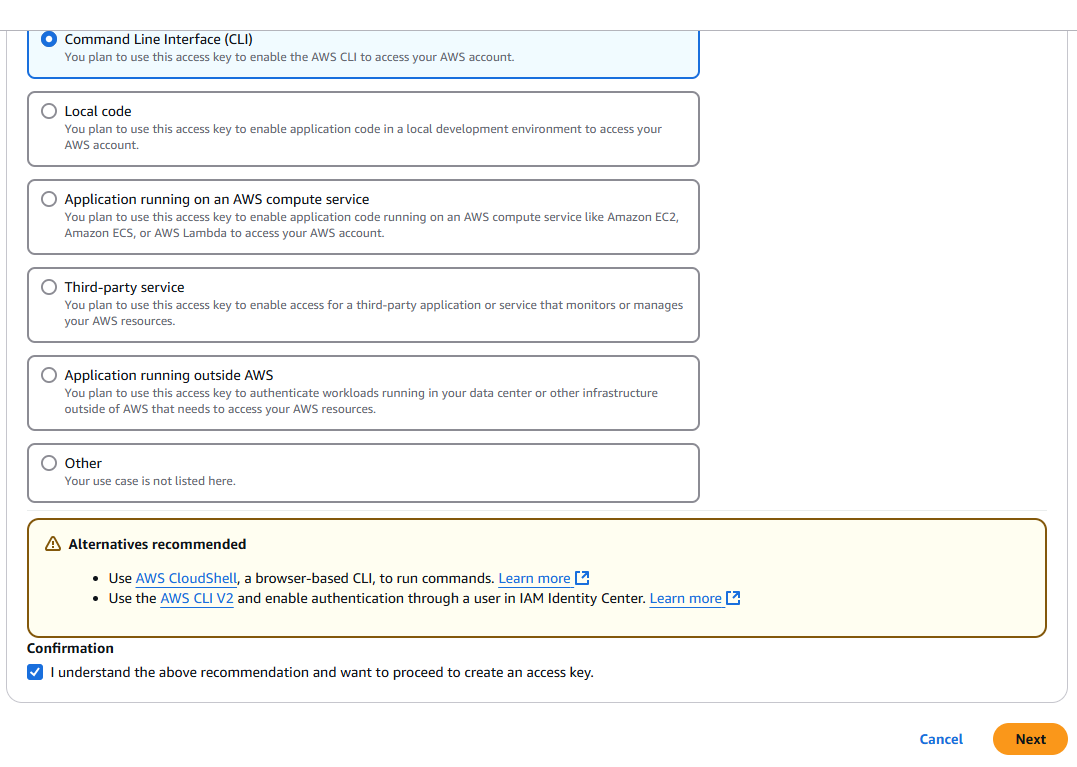
Now, Create access key



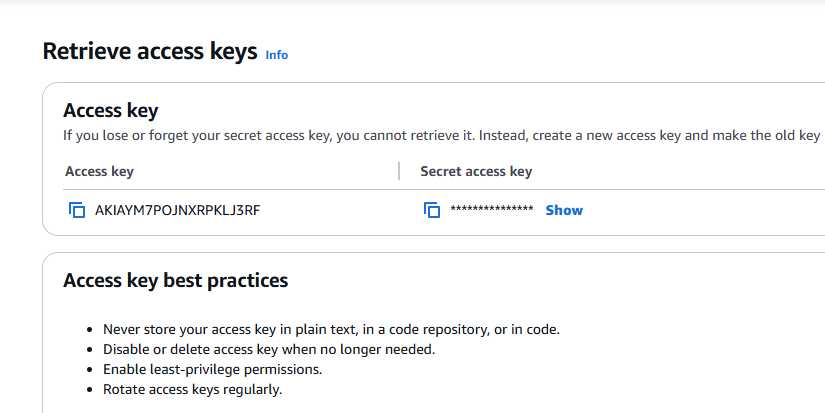


Application running outside AWS means SDK

CLI



We need both Security access key and Access key for CLI access



We have understood about User, UserGroups, Policies --> AWS managed policies, Customer managed policies, Roles -->