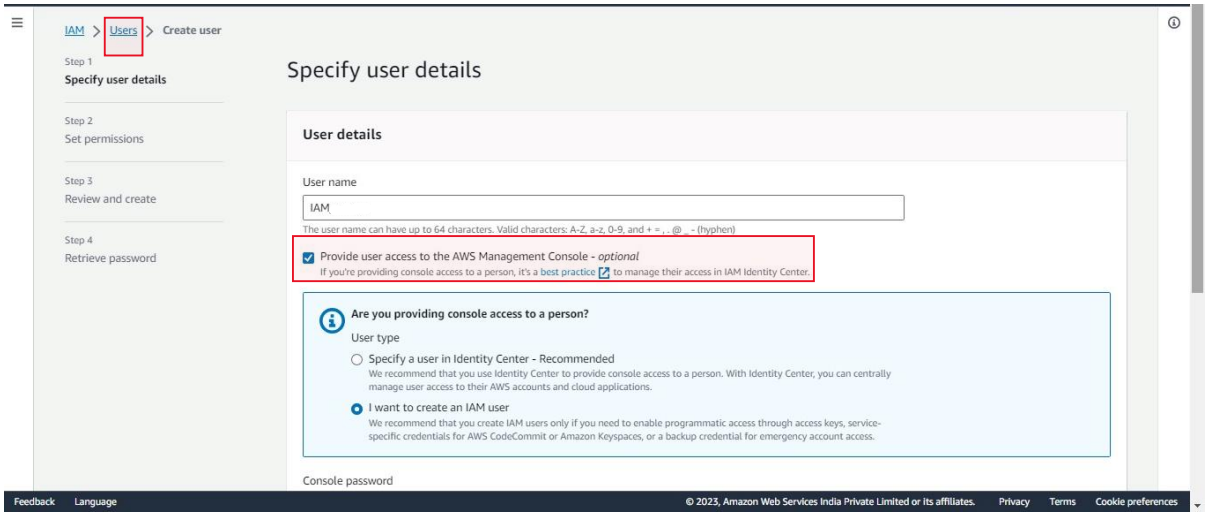
IAM AWS - Create a User

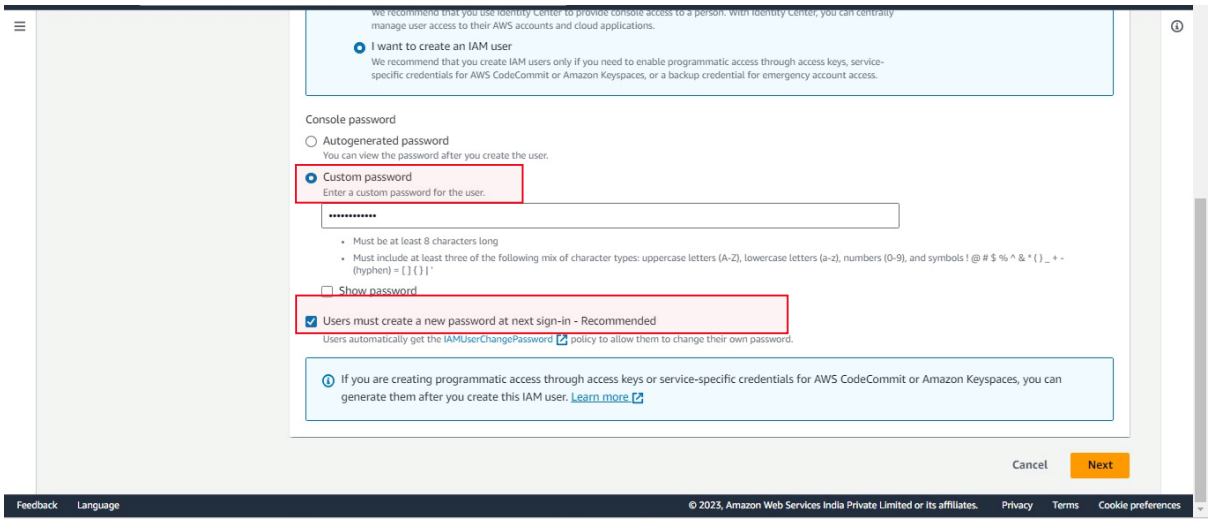
Objectives

- As you probably know at this point, it is not recommended to work with the root account in AWS.
- For this reason, we are going to create a new account which we will use regularly as the admin account.

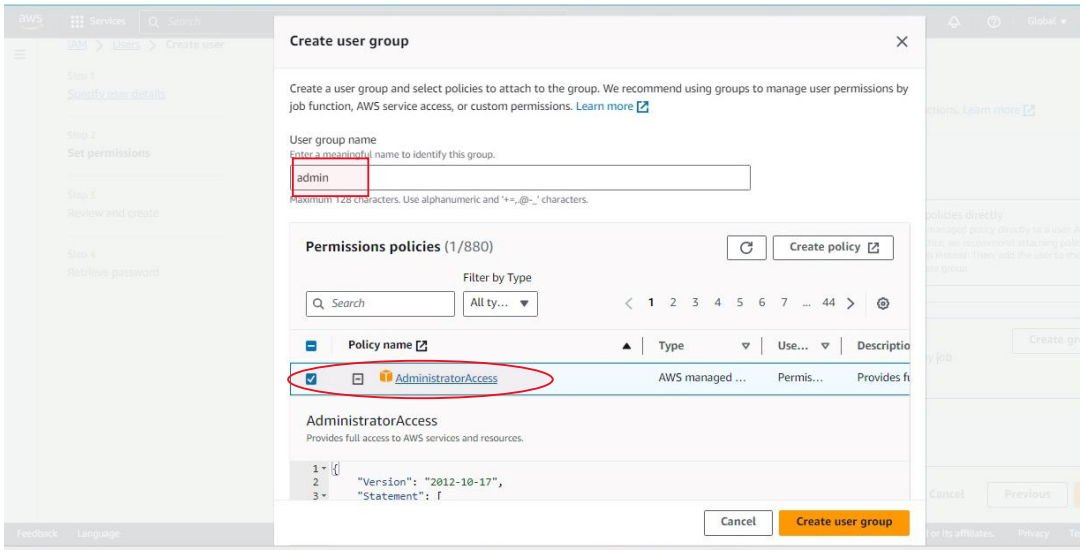
1. Create an IAM user with password credentials.
2. Add the newly created user to a group called "**admin**" and attach to it the policy called "**Administrator Access**"
3. Make sure the user has a tag called with the key Role and the value DevOps.



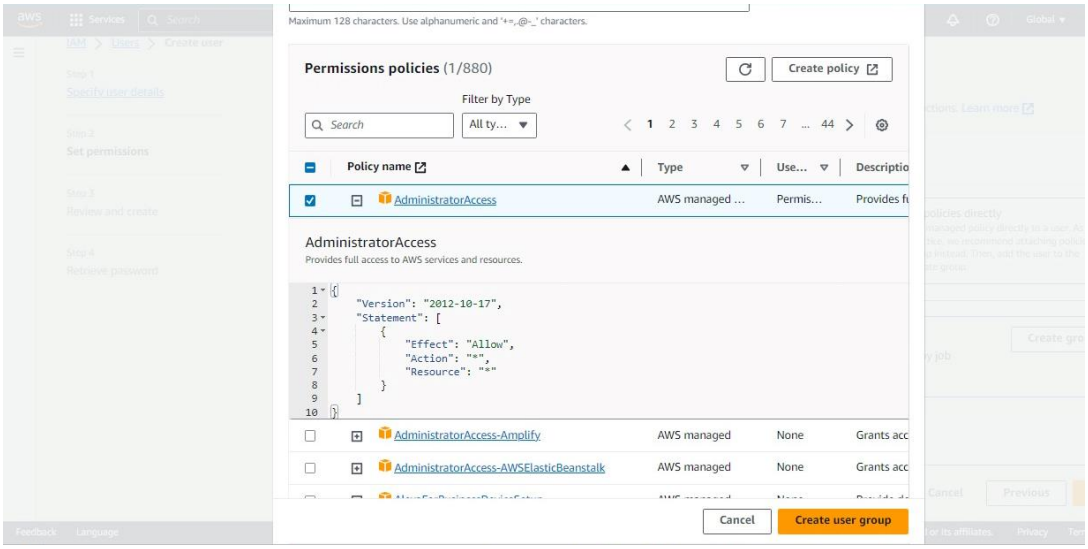
- Give the username and tick on the option **"provide user access to AWS Management Console."**
- Select the Option **"I want to create an IAM user."**



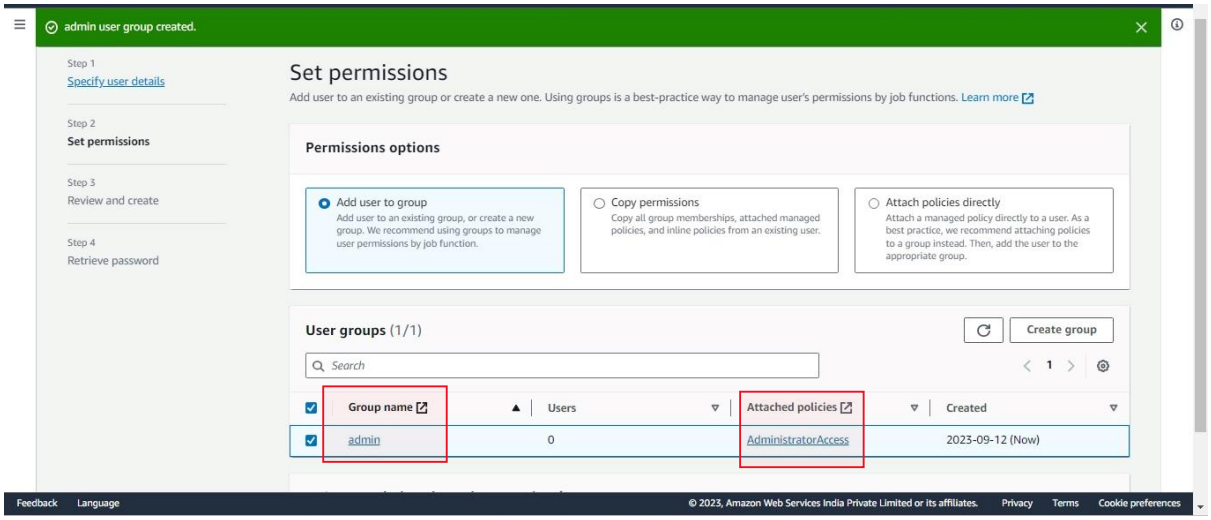
- Create a custom password and give the password of your choice.
- If you want you can tick on the option of users can create a new password on next sign-in.



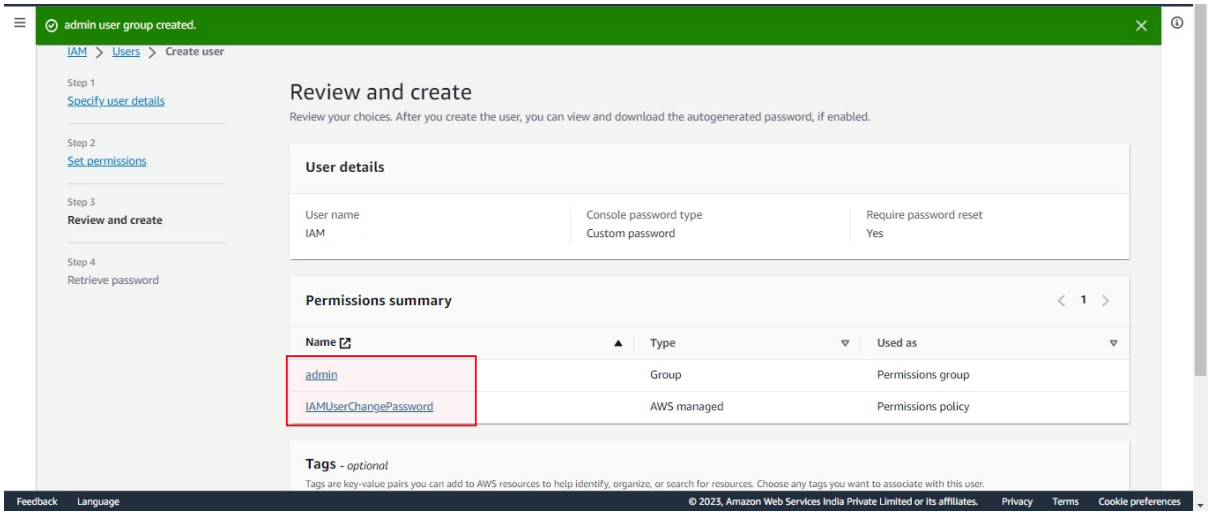
- In the set permissions section, you will get an option add user to group click on create group.
- Give the group name as "admin" and add **"administrator access"** policy to it so that user can perform all the activities.



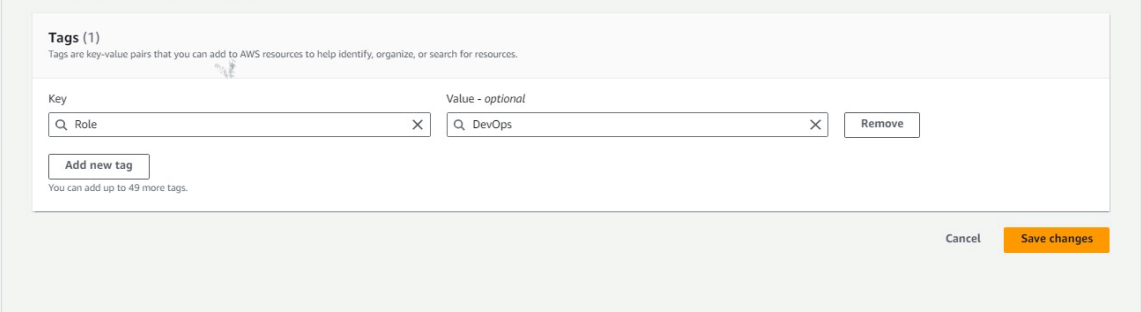
Click on create user group.



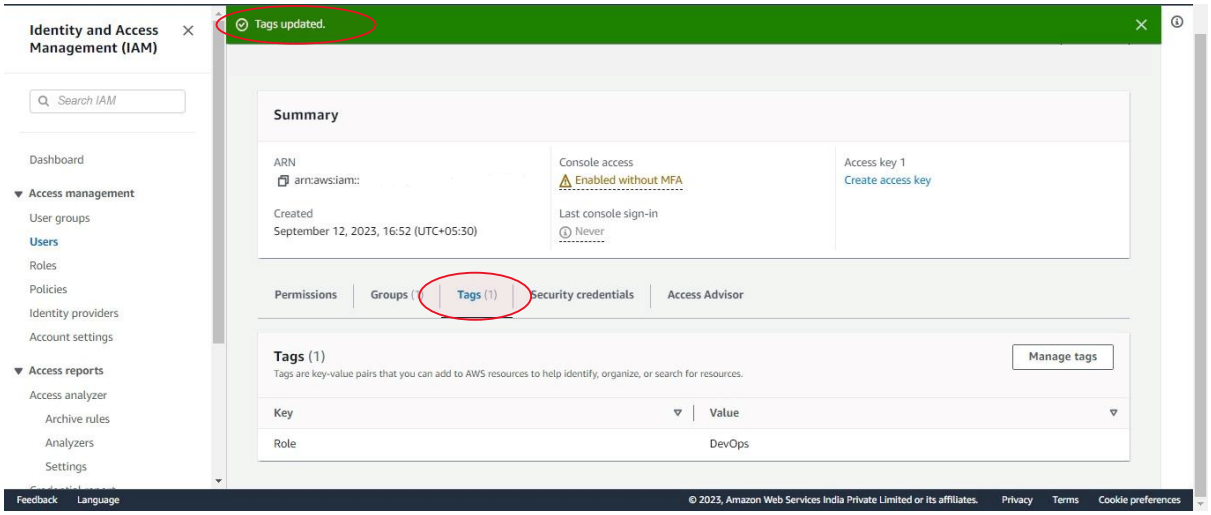
Group created successfully also we can see the attached policies to the group.



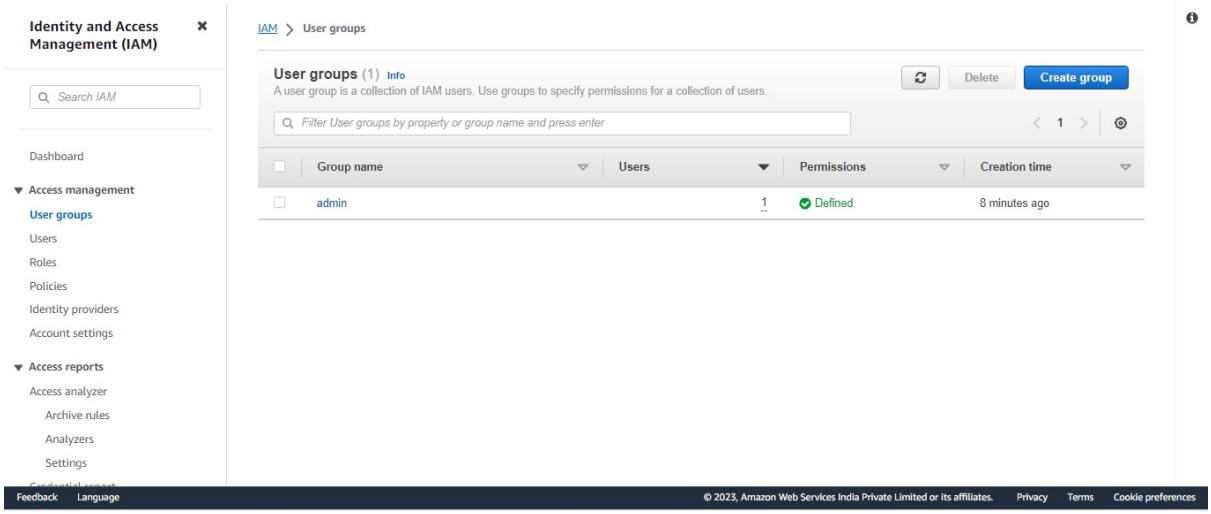
Review the details of the user as we can see following permissions has been given to the user.



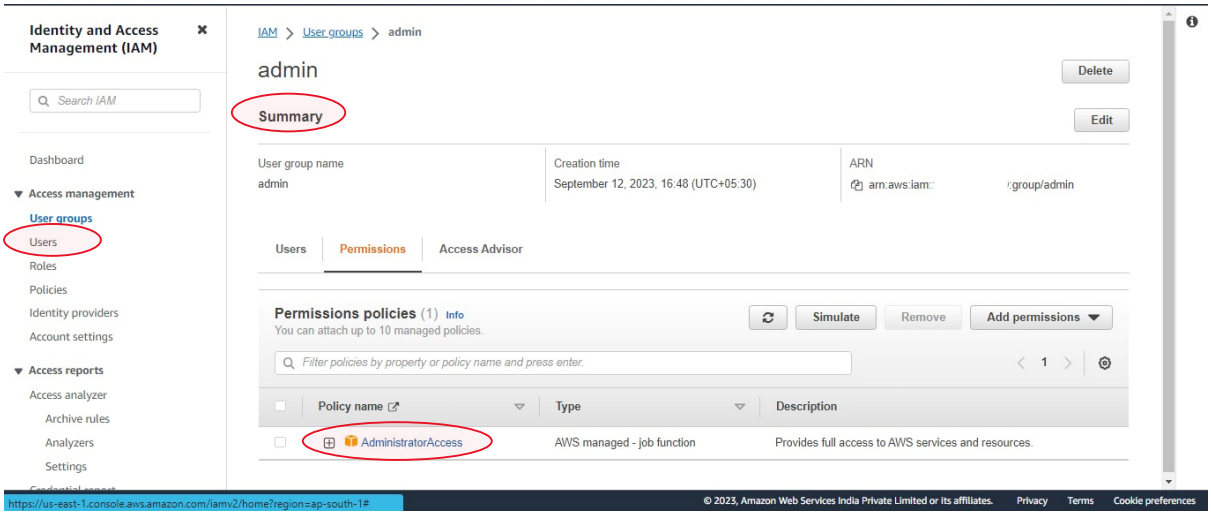
Give the following tags to the user.



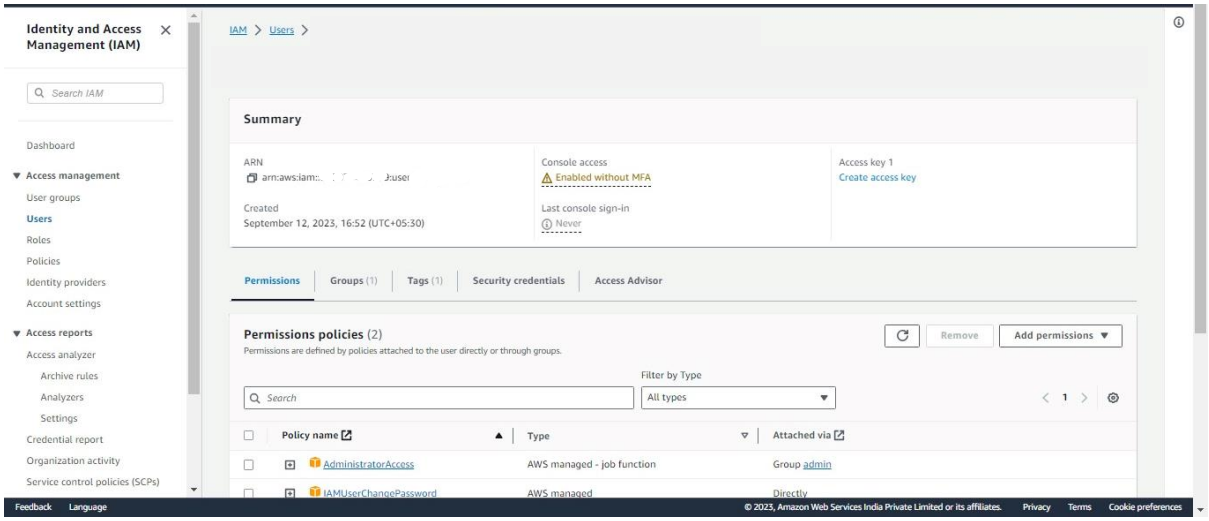
Tags updated successfully.



In the user groups section, we can see the user groups that we have created.



Following is the summary of the user.



Following is the ARN of the user.

Solution

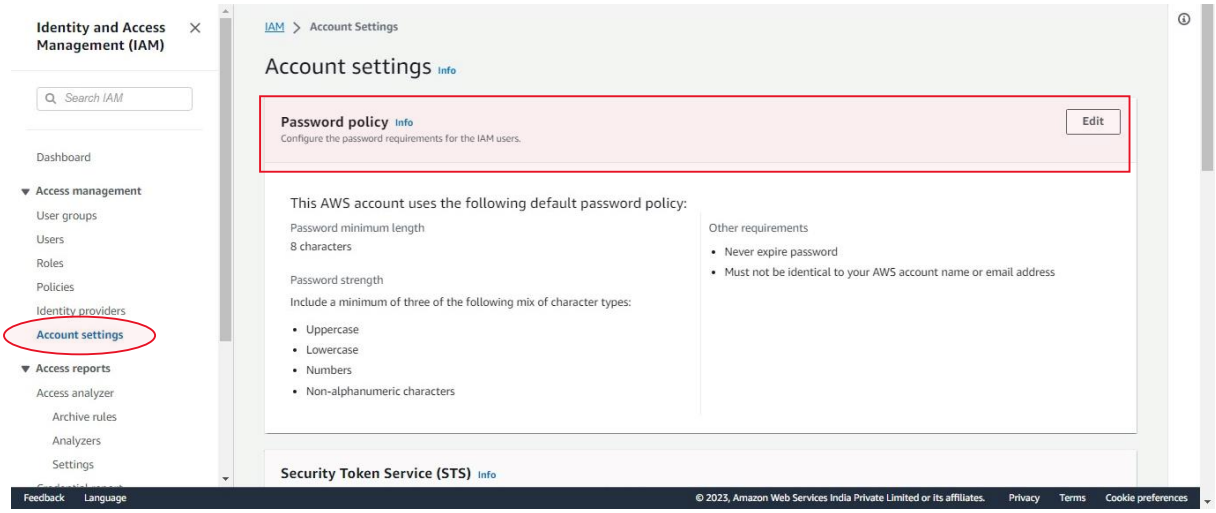
Steps to create a user.

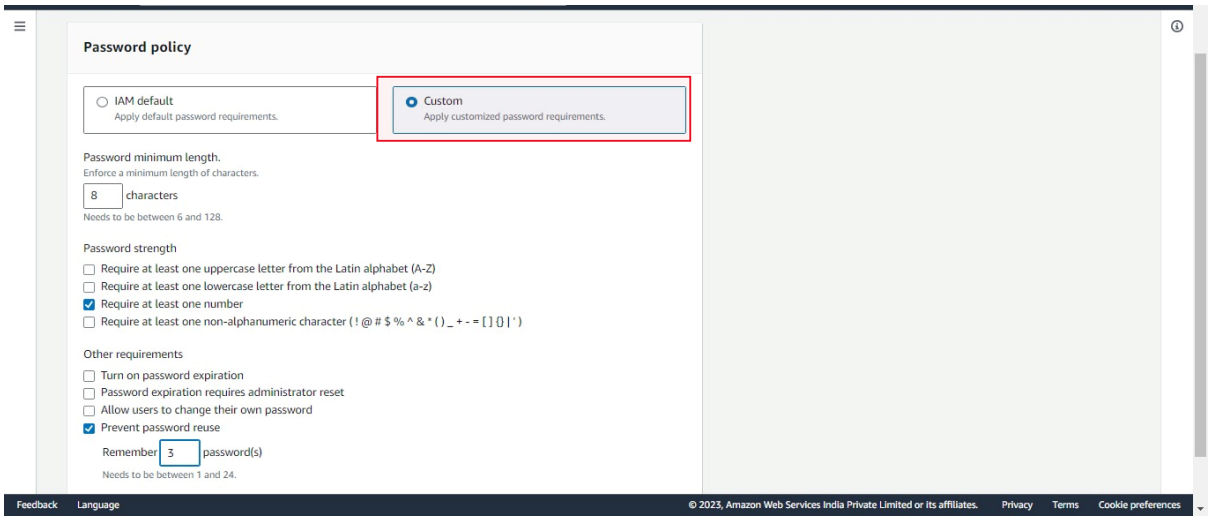
1. Go to the AWS IAM service
2. Click on "Users" in the right-side menu (right under "Access Management")
3. Click on the button "Add users"
4. Insert the user's name (e.g. Mario)
5. Select the credential type: "Password"
6. Set console password to custom and click on "Next"
7. Click on "Add user to group"
8. Insert "admin" as group name
9. Check the "Administrator Access" policy and click on "Create group"
10. Click on "Next: Tags"
11. Add a tag with the key Role and the value DevOps
12. Click on "Review" and then create on "Create user"

AWS IAM - Password Policy & MFA

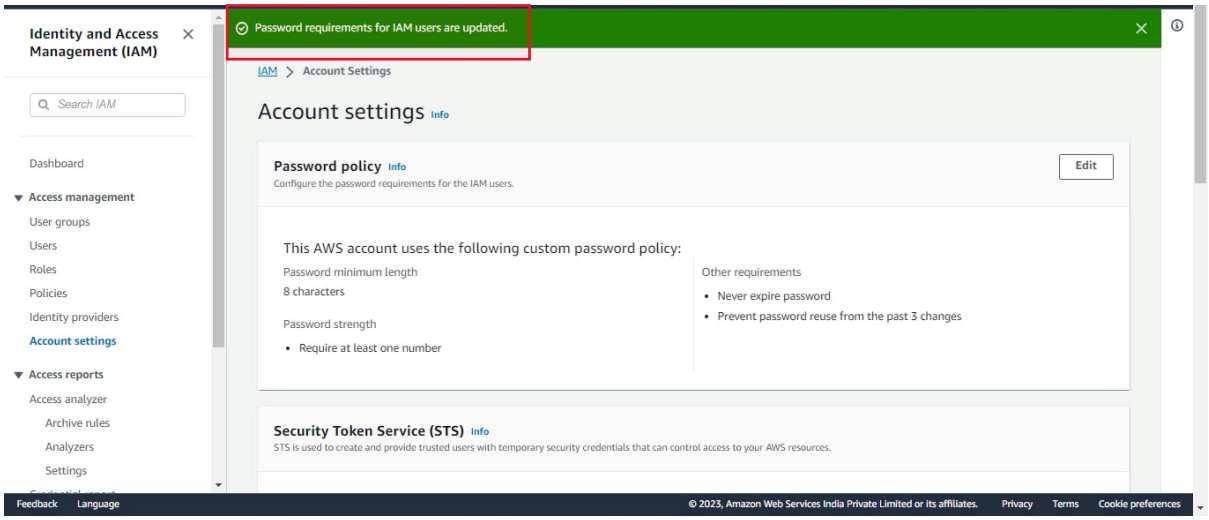
Objectives

1. Create password policy with the following settings:
2. At least minimum 8 characters.
3. At least one number.
4. Prevent password reuse.
5. Then enable MFA for the account.



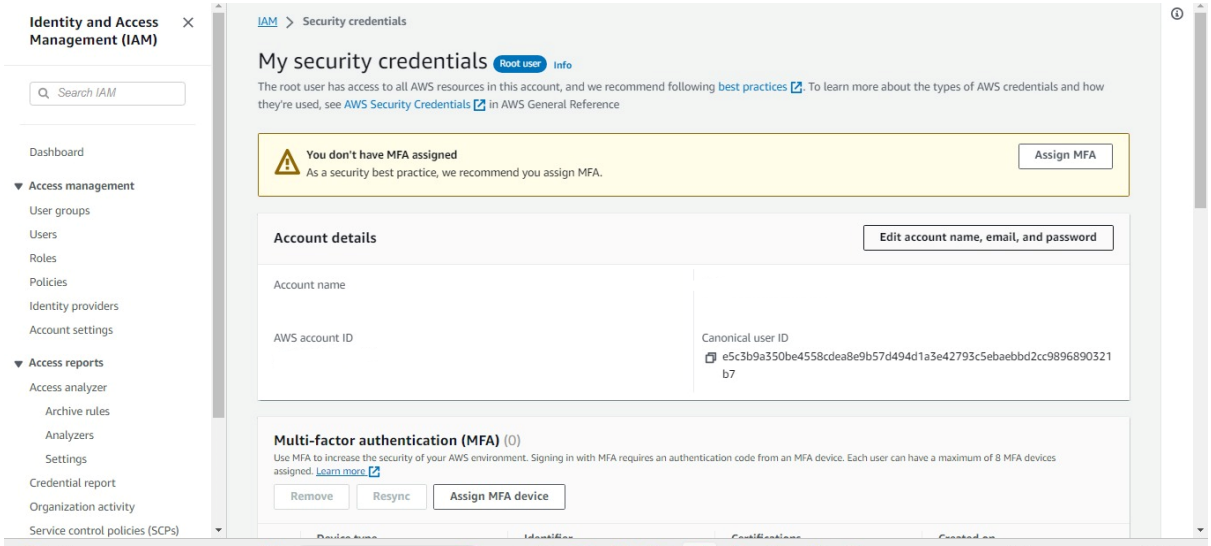


Set the conditions for the password tick on the checkboxes of the conditions you want.

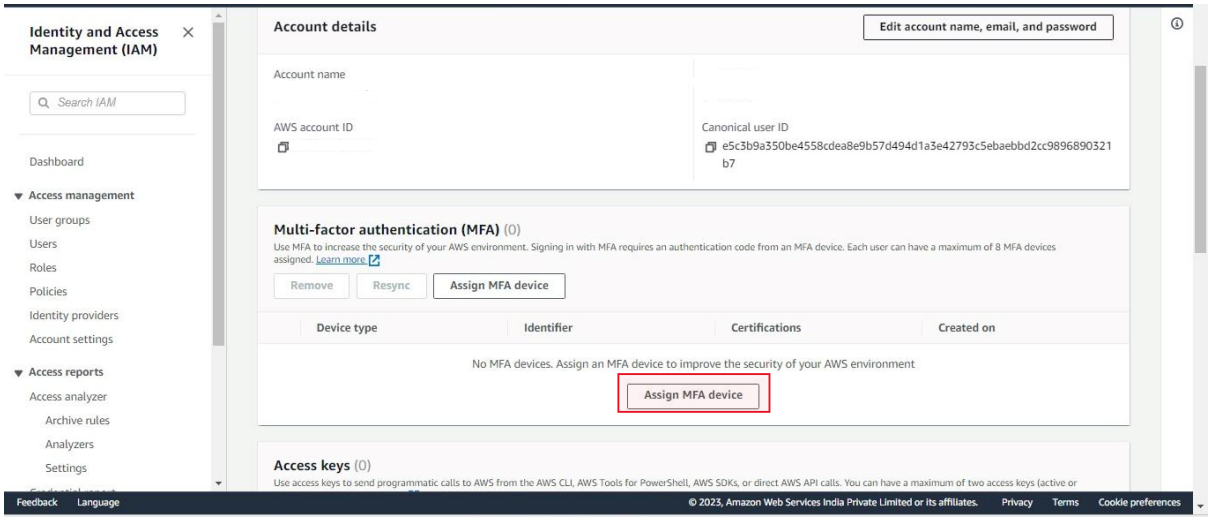


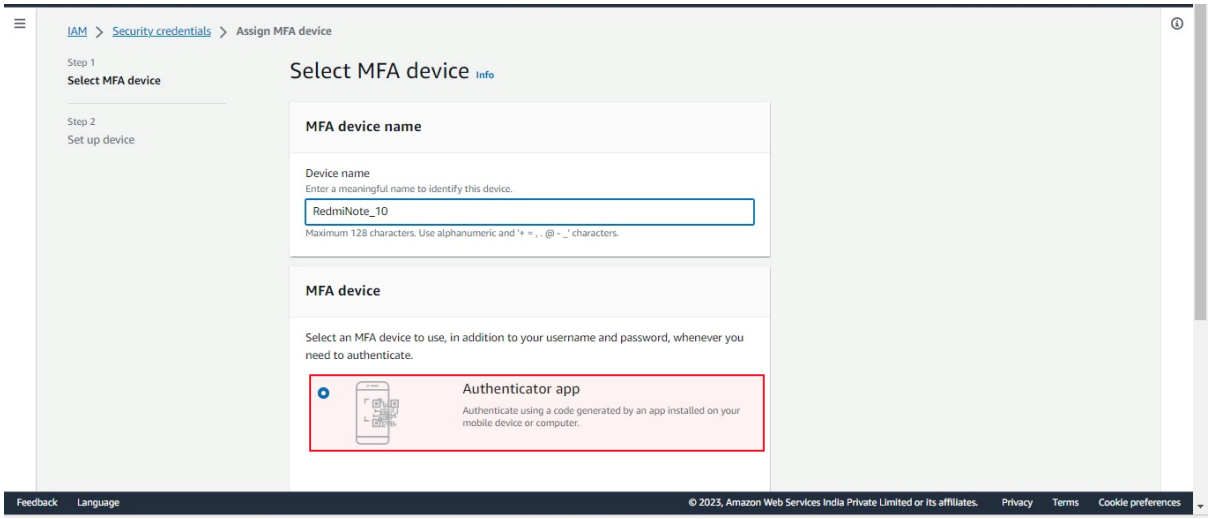
As we cans see password requirements for the IAM users has been set successfully.

For Multi-Factor Authentication



- Under security credentials section you can assign MFA to your account.
- For security best practices we must assign MFA.



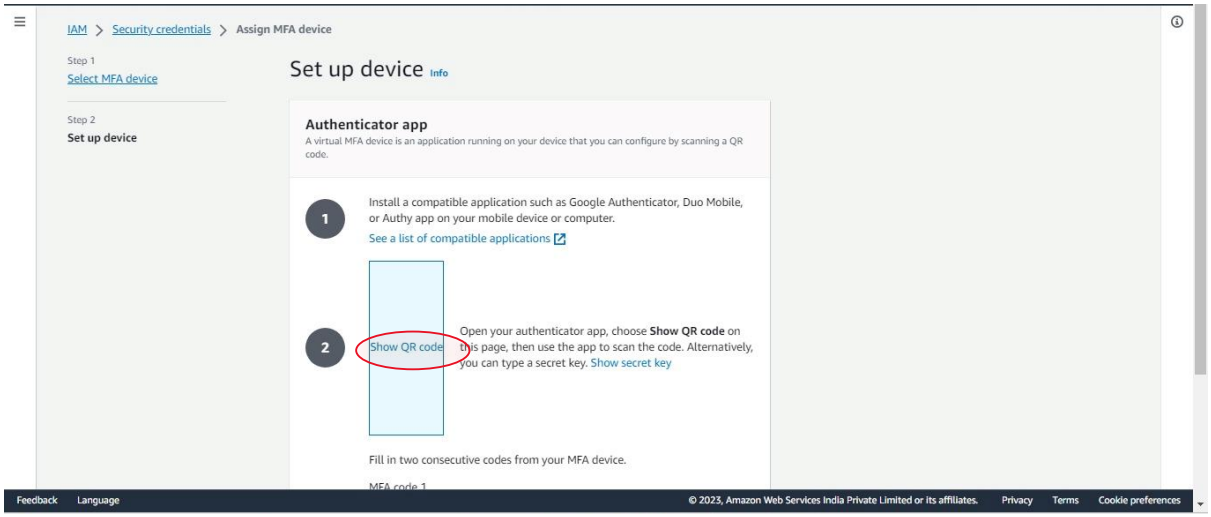


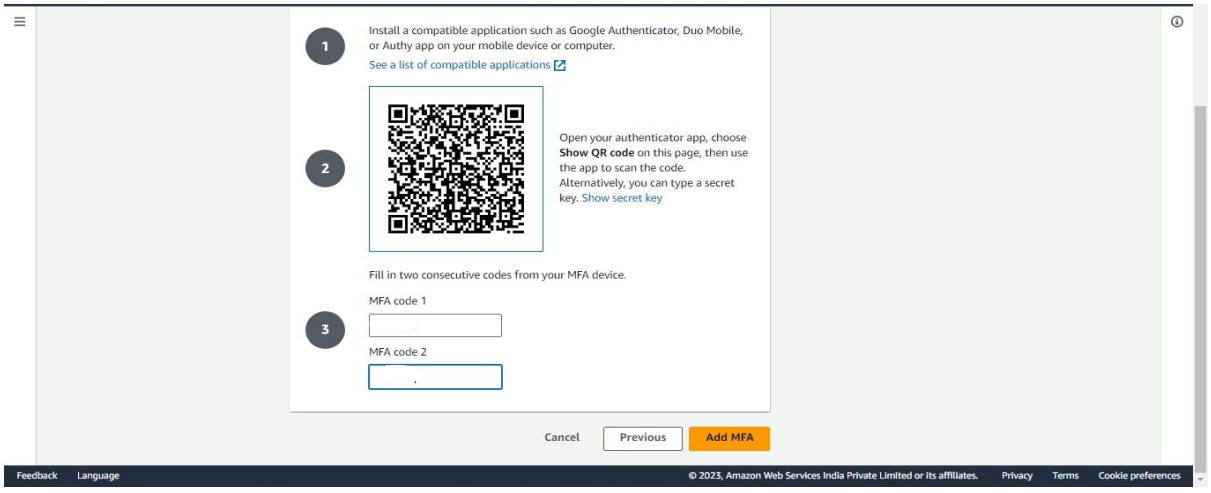
Set up your first account

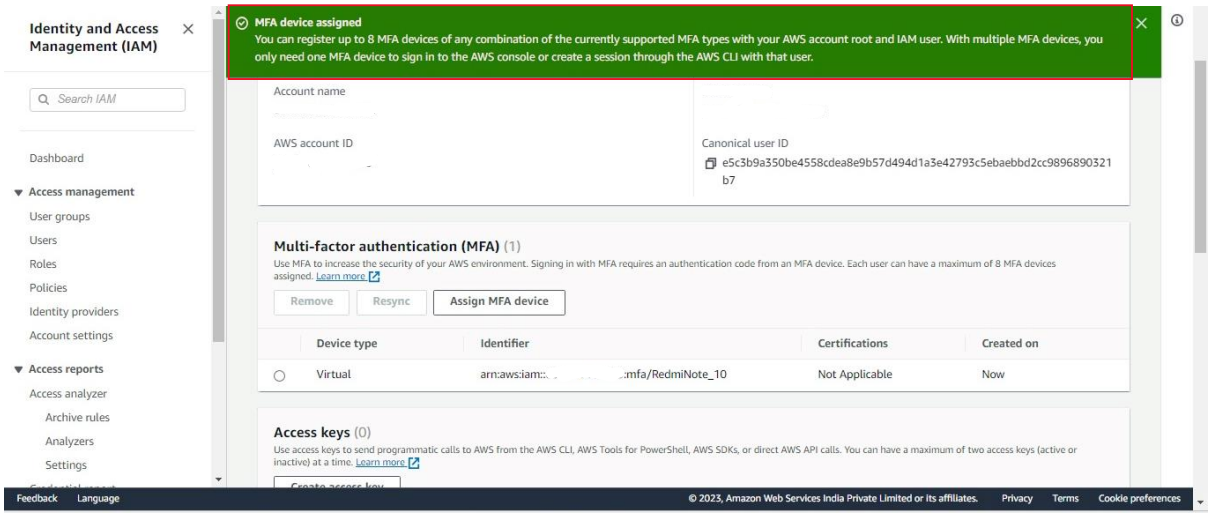
Use the QR code or setup key in your 2FA settings (by Google or third-party service). If you're having trouble, go to g.co/2sv

Scan a QR code

Enter a setup key







Solution

Password Policy:

1. Go to IAM service in AWS
2. Click on "Account settings" under "Access management"
3. Click on "Change password policy"
4. Check "Enforce minimum password length" and set it to 8 characters
5. Check "Require at least one number"
6. Check "Prevent password reuse"
7. Click on "Save changes"

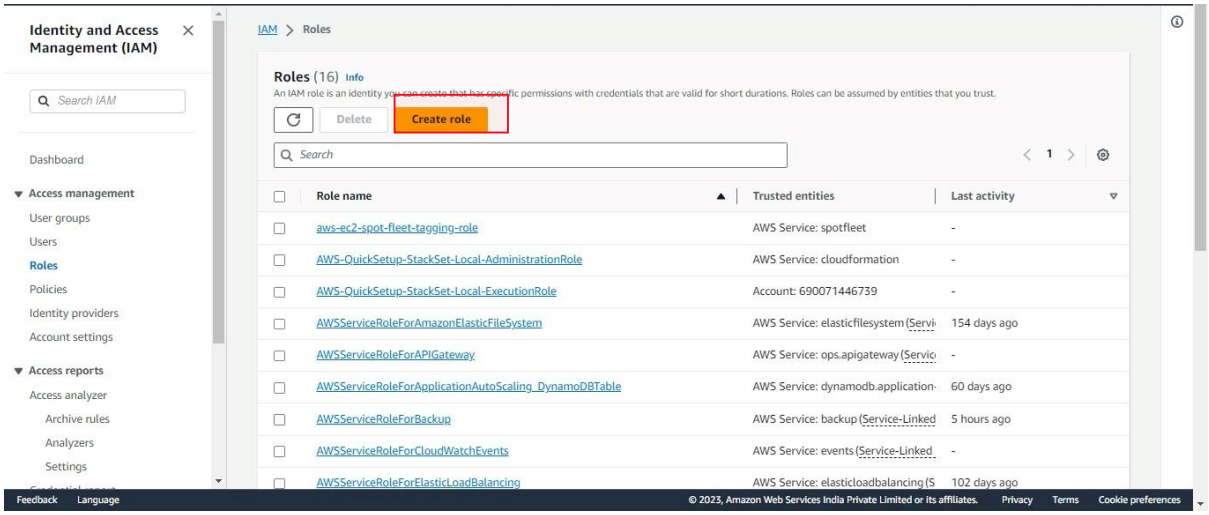
MFA:

1. Click on the account name
2. Click on "My Security Credentials"
3. Expand "multi-factor authentication (MFA)" and click on "Activate MFA"
4. Choose one of the devices
5. Follow the instructions to set it up and click on "Assign MFA"

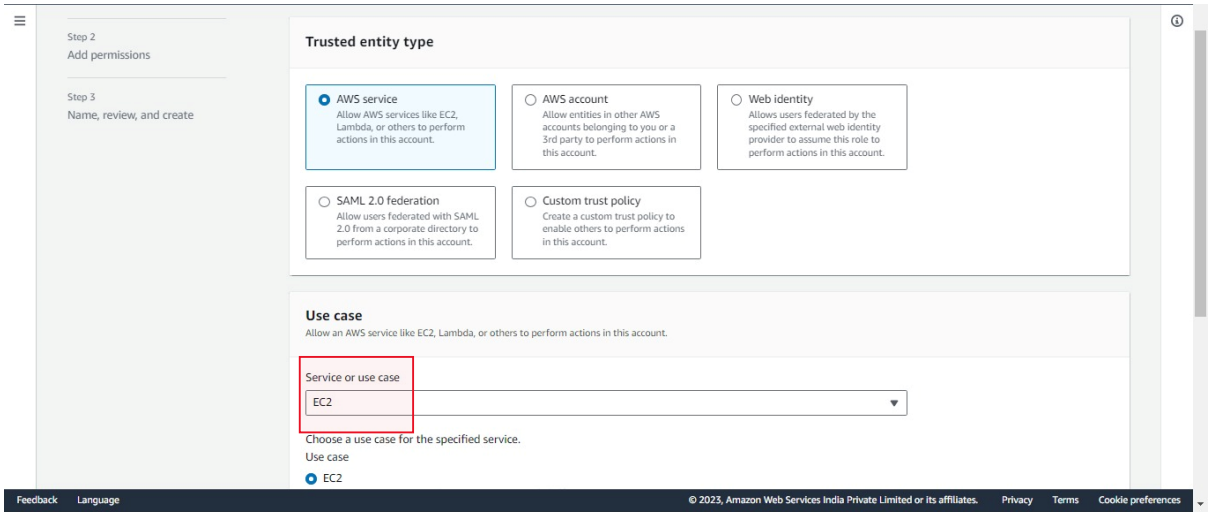
AWS - Create a Role

Objectives

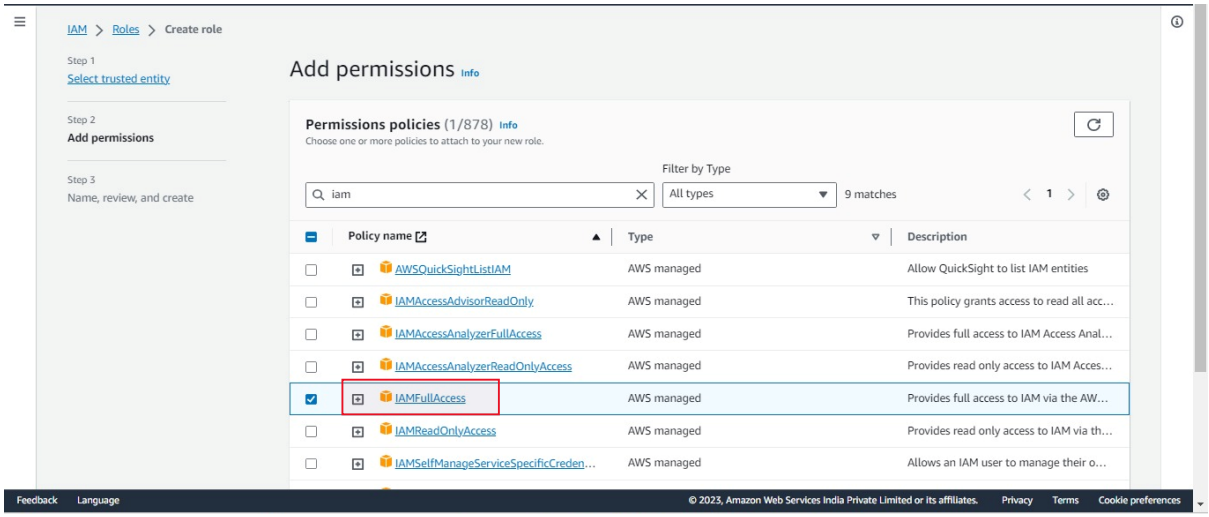
Create a basic role to provide EC2 service with Full IAM access permissions. In the end, run from the CLI (or Cloud Shell) the command to verify the role was created.



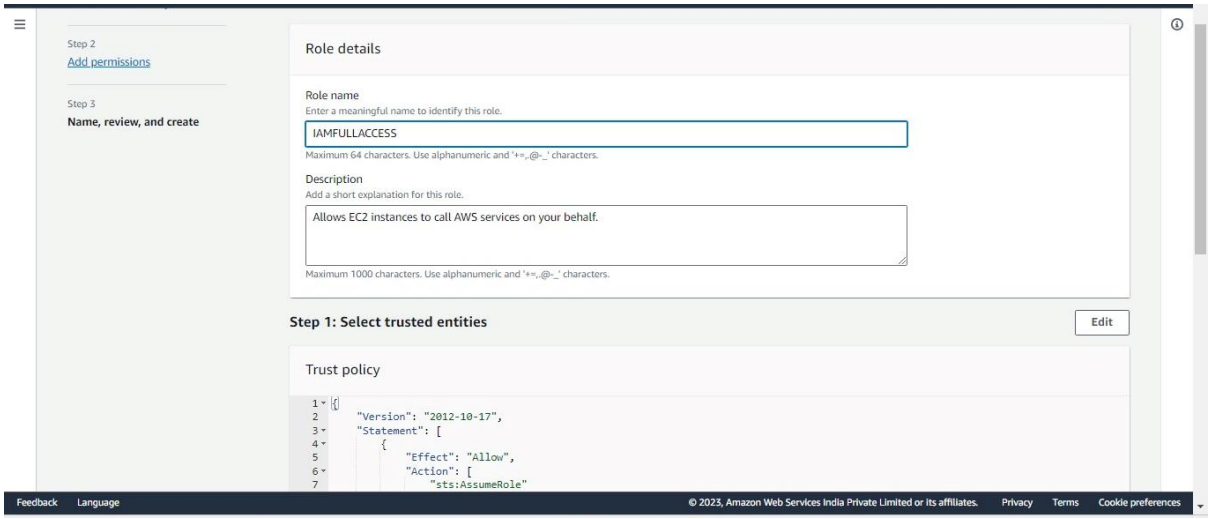
In IAM console under roles section click on create role.



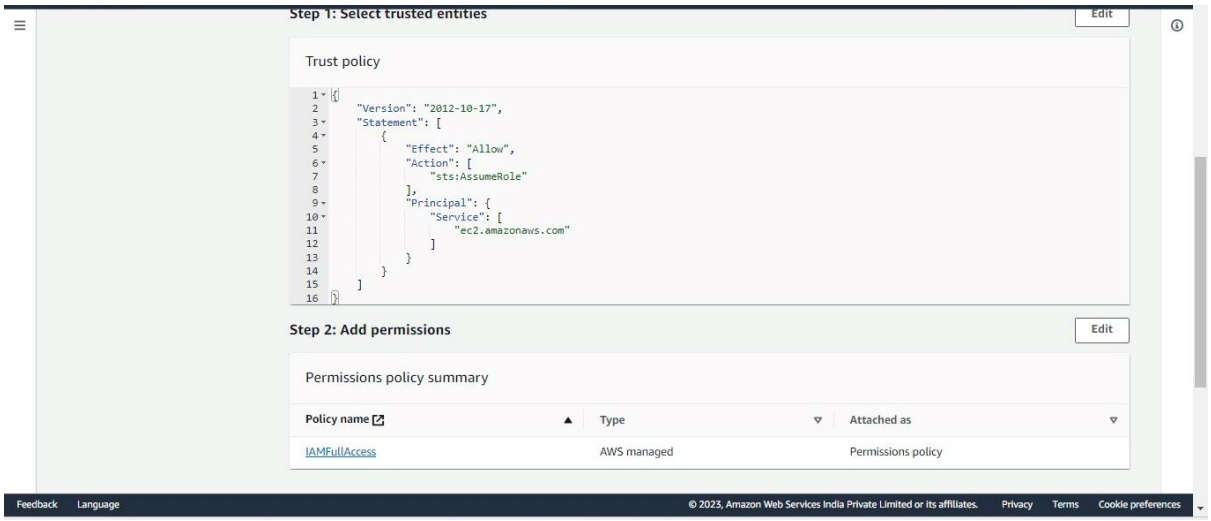
Choose the use case as EC2.

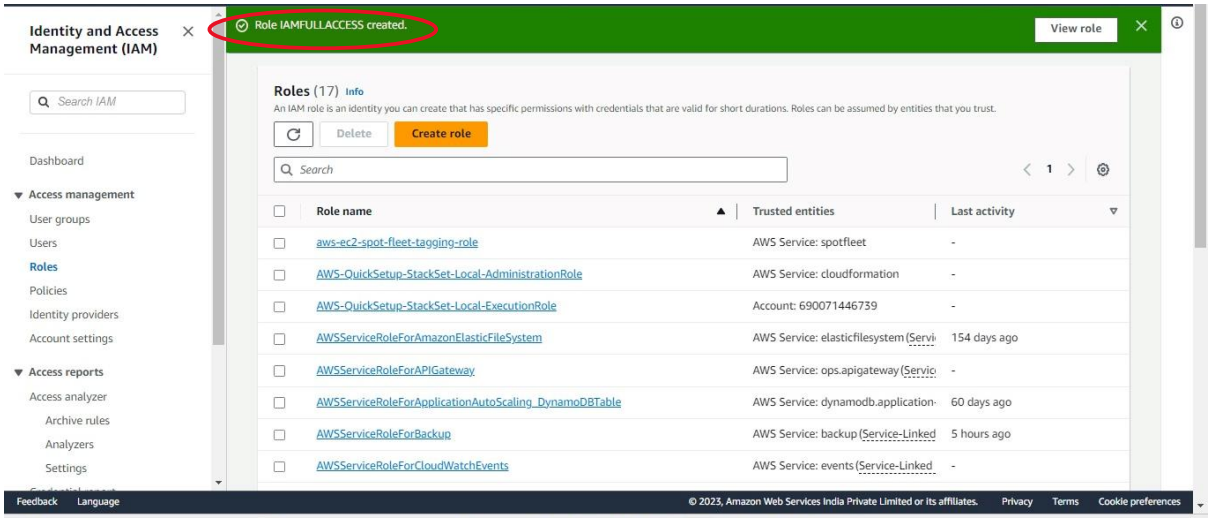


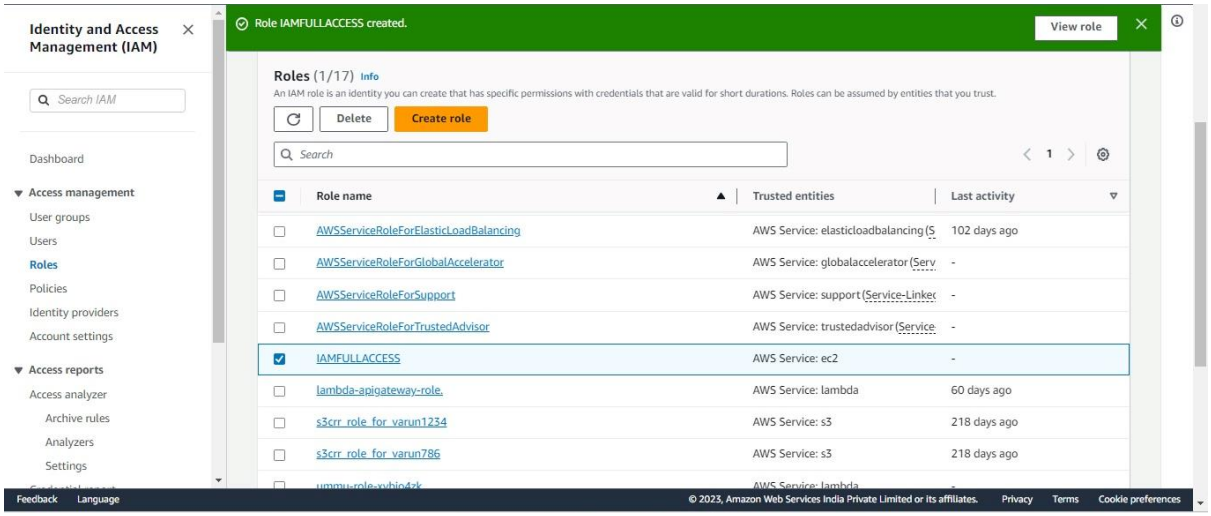
Give the permission policy to the role as IAM Full access.



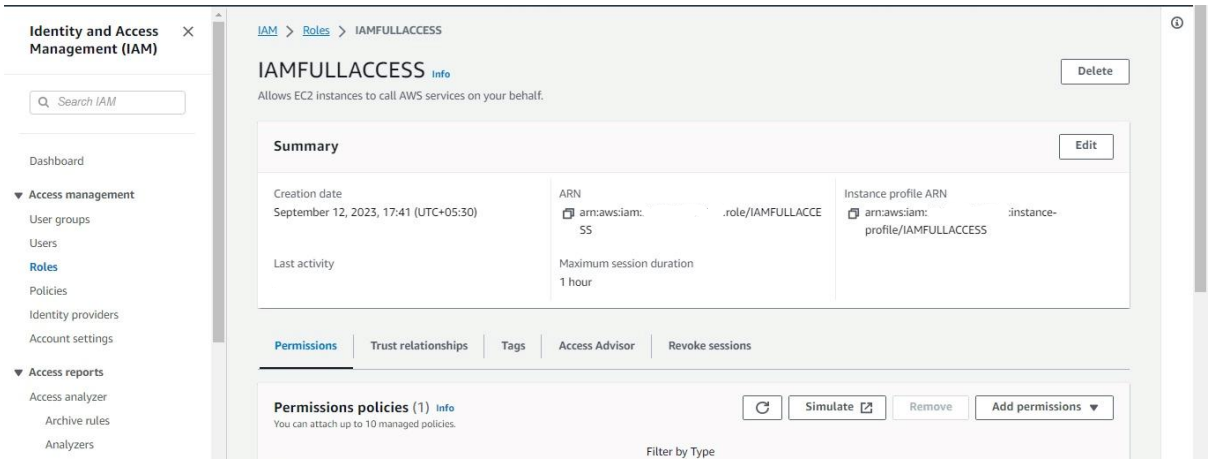
Give the Role name and add description.







As we can see our role has been successfully created and we have given the permission of IAM Full Access to this role.



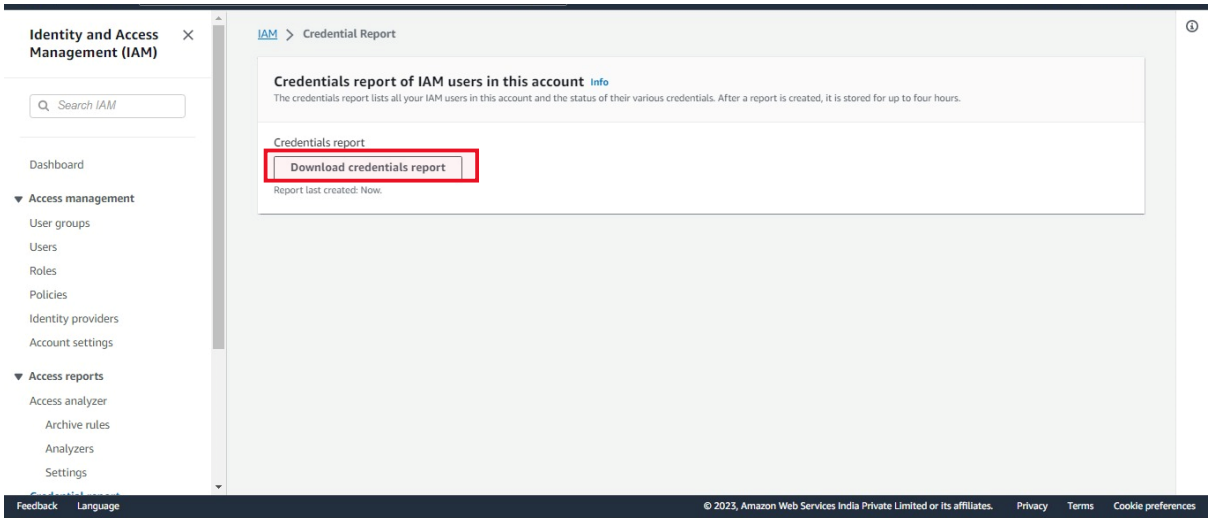
Solution

1. Go to AWS console -> IAM
2. Click in the left side menu on "Access Management" -> Roles
3. Click on "Create role"
4. Choose "AWS service" as the type of trusted entity and then choose "EC2" as a use case. Click on "Next"
5. In permissions page, check "IAMFullAccess" and click on "Next" until you get to "Review" page
6. In the "Review" page, give the role a name (e.g. IAMFullAcessEC2), provide a short description and click on "Create role"
7. aws iam list-roles will list all the roles in the account, including the one we've just created.

AWS - Credential Report

Objectives

1. Create/Download a credential report
2. Answer the following questions based on the report:
3. Are there users with MFA not activated?
4. Explain the use case for using the credential report?



Solution

1. Go to the AWS IAM service
2. Under "Access Reports" click on "Credential report"
3. Click on "Download Report" and open it once it is downloaded
4. Answer the questions in these exercises by inspecting the report

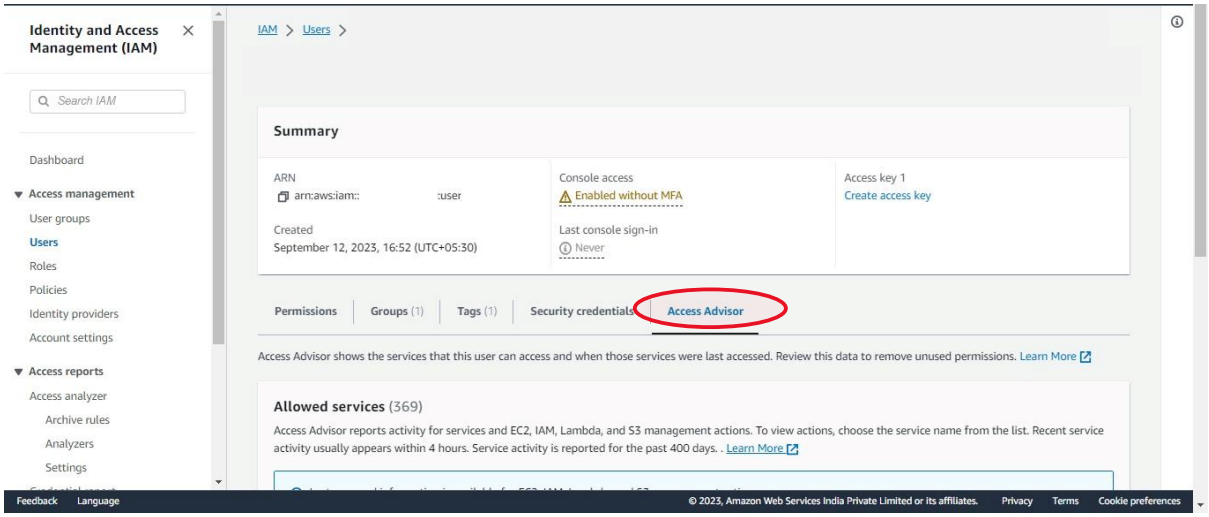
- The credential report is useful to identify whether there any users who need assistance or attention in regards to their security.
- For example, a user who did not change his password for a long time and did not activate MFA.

AWS IAM - Access Advisor

Objectives

Go to the Access Advisor and answer the following questions regarding one of the users:

1. Are there services this user never accessed?
2. What was the last service the user has accessed?
3. What the Access Advisor is used/good for?



Solution

1. Go to AWS IAM service and click on "Users" under "Access Management"
2. Click on one of the users
3. Click on the "Access Advisor" tab
4. Check which service was last accessed and which was never accessed

Access Advisor can be good to evaluate whether there are services the user is not accessing (as in never or not frequently). This can be help in deciding whether some permissions should be revoked or modified.