**IAM (Identity Access Management):**

IAM is a service which is used to access AWS services and resources. IAM is completely free for life time.

In IAM,

**Identity:** Authentication

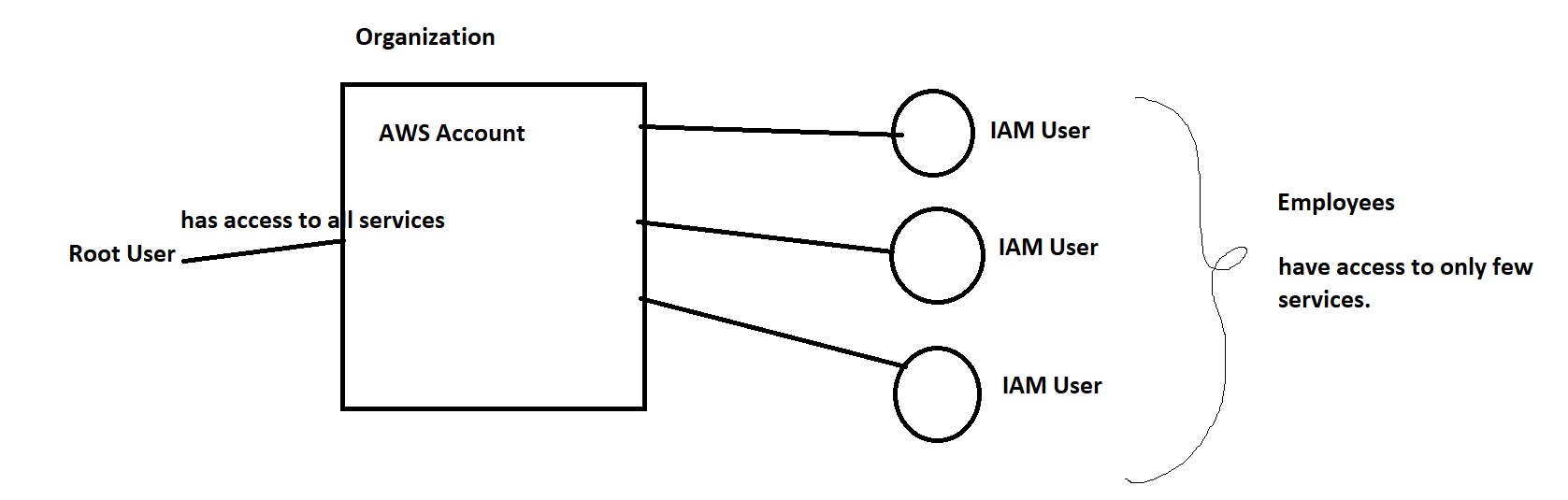
**Access:** Authorization

* Authentication is just for sign in to AWS account.
* Authorization is providing access (permissions) to a particular service.

Using IAM we can also place “Multi-factor Authentication [MFA]” for the root user.

**In AWS we have 2 types of Users.**

1. **Root User:** The user who created AWS account and has access on all 200+ services
2. **IAM User:** IAM user is created by root user. IAM user can able to sign in to AWS account. But don’t have access/ permissions to use any service.



Here, DB team has access to only few services, DevOps team has access to only DevOps

**Sub services in IAM:**

1. Users
2. Policies
3. Groups
4. Role

**IAM user:**

* IAM user is a user who is created by root user to access only specific services from the AWS account.
* He/ She don’t have permissions to all the services, these permissions are provided by the root user.
* We can provide 2 types of access to the IAM user.

1. AWS console access
2. Programmatical access

**IAM Policies:**

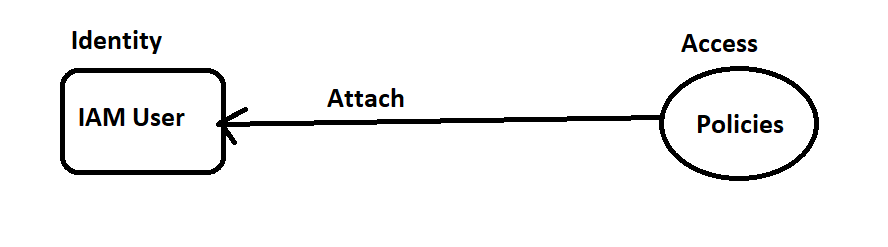
* IAM Policy means set of permissions that are added to a particular IAM user or Group.
* If you are an IAM User, you can able to login but don’t have access to any service to access any service attach policies for IAM user then only he/she can access the service.
* There are 2 types of IAM Policies

1. AWS managed

Ex: AdminFullAccess , S3FullAccess, S3ReadOnly etc

1. Customer managed policies:

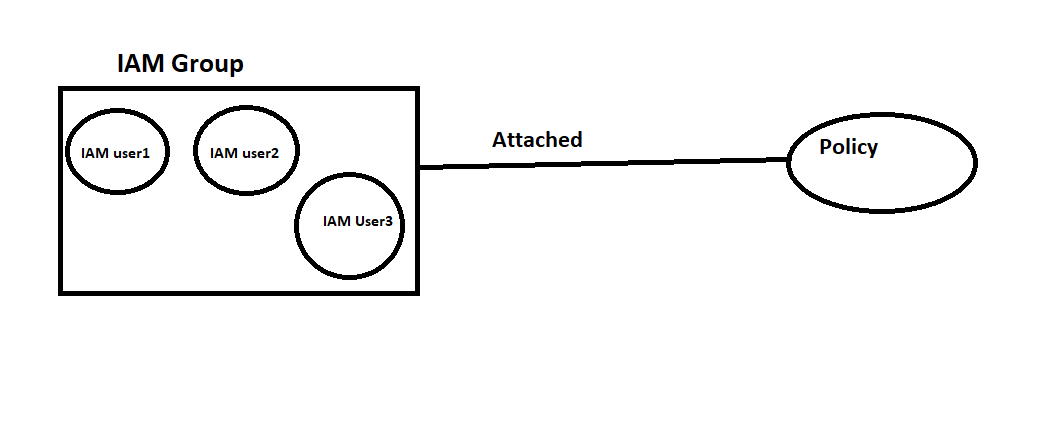
You can write your own policies using Json. This is called Inline Policy.



**IAM Groups:**

Suppose I want to assign same policies to a particular group of people (IAM users). Then I can create a IAM group and attach the policies to that group and add IAM users to that group.

So, that I can attach the policy at once and it reflects to all the users in that IAM group.



**IAM Role:**

Suppose, I want to provide temporary access to any service which can be automatically expired after 1 hour then I should create a IAM Role.

IAM role doesn’t have access to AWS console. It has only access key and secret key when you want to perform a task.

**Creating an IAM User and adding policies:**

If you are a root user then you have access to IAM. Or if you are an IAM user then you need to get IAM full access to create IAM user.

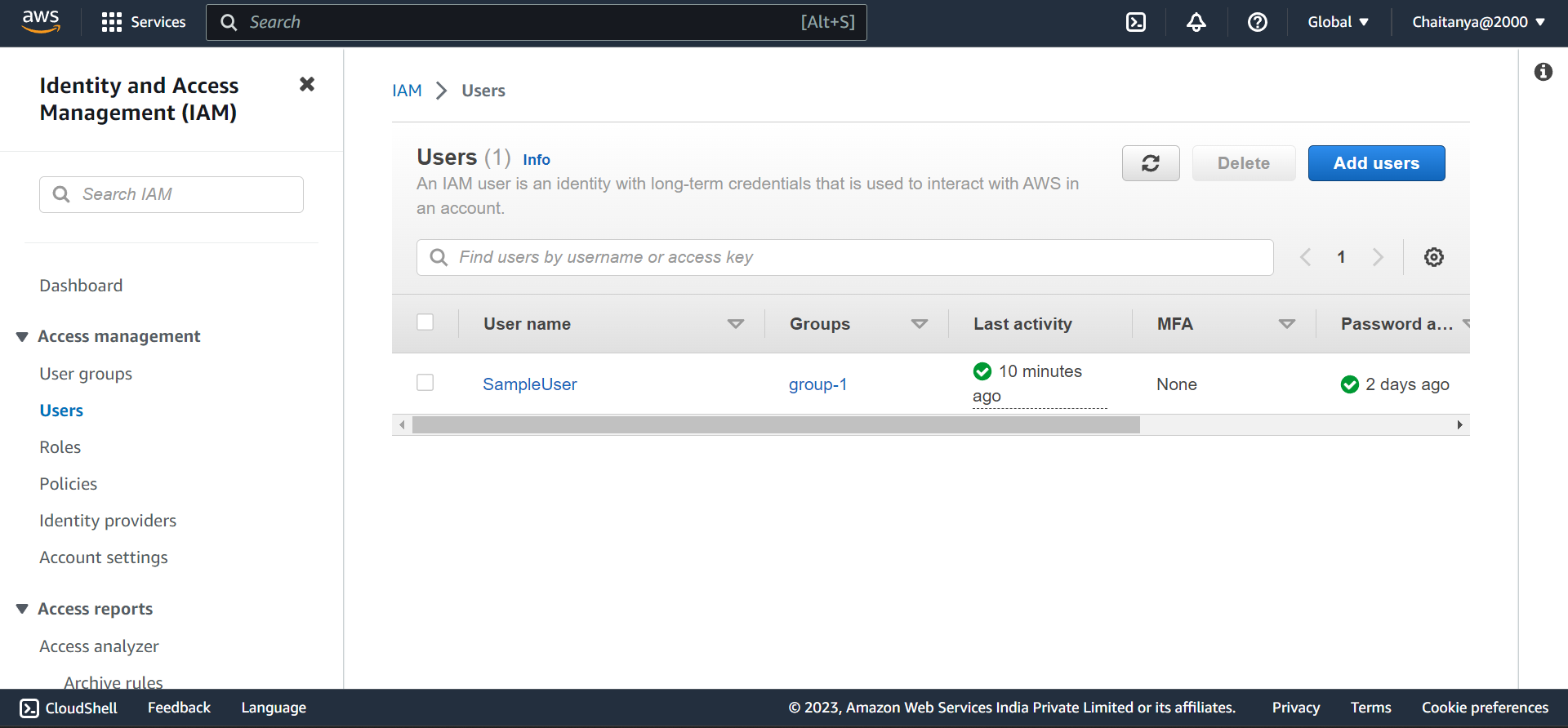
Steps:

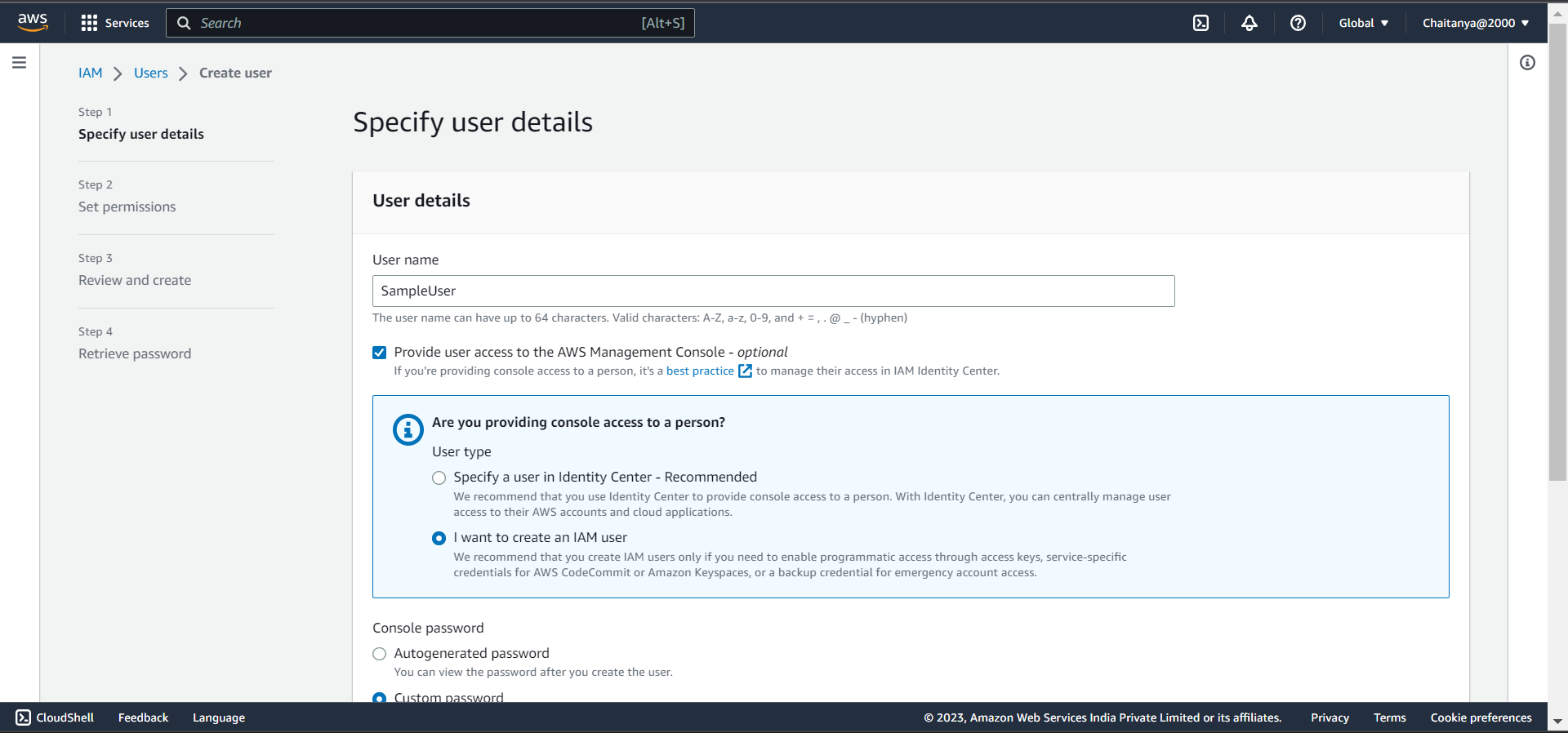
1. Login to AWS console.
2. Search for IAM
3. Navigate to IAM dashboard
4. Select users in the access management from left panel
5. You can see existing IAM users if created.
6. Click on Add users
7. You can provide 2 types of access.
8. AWS Console access
9. Programmatical access
10. Select AWS console access and select I want to create an IAM user
11. Here, I want you can choose an autogenerated password or create a custom password.
12. Click on Next

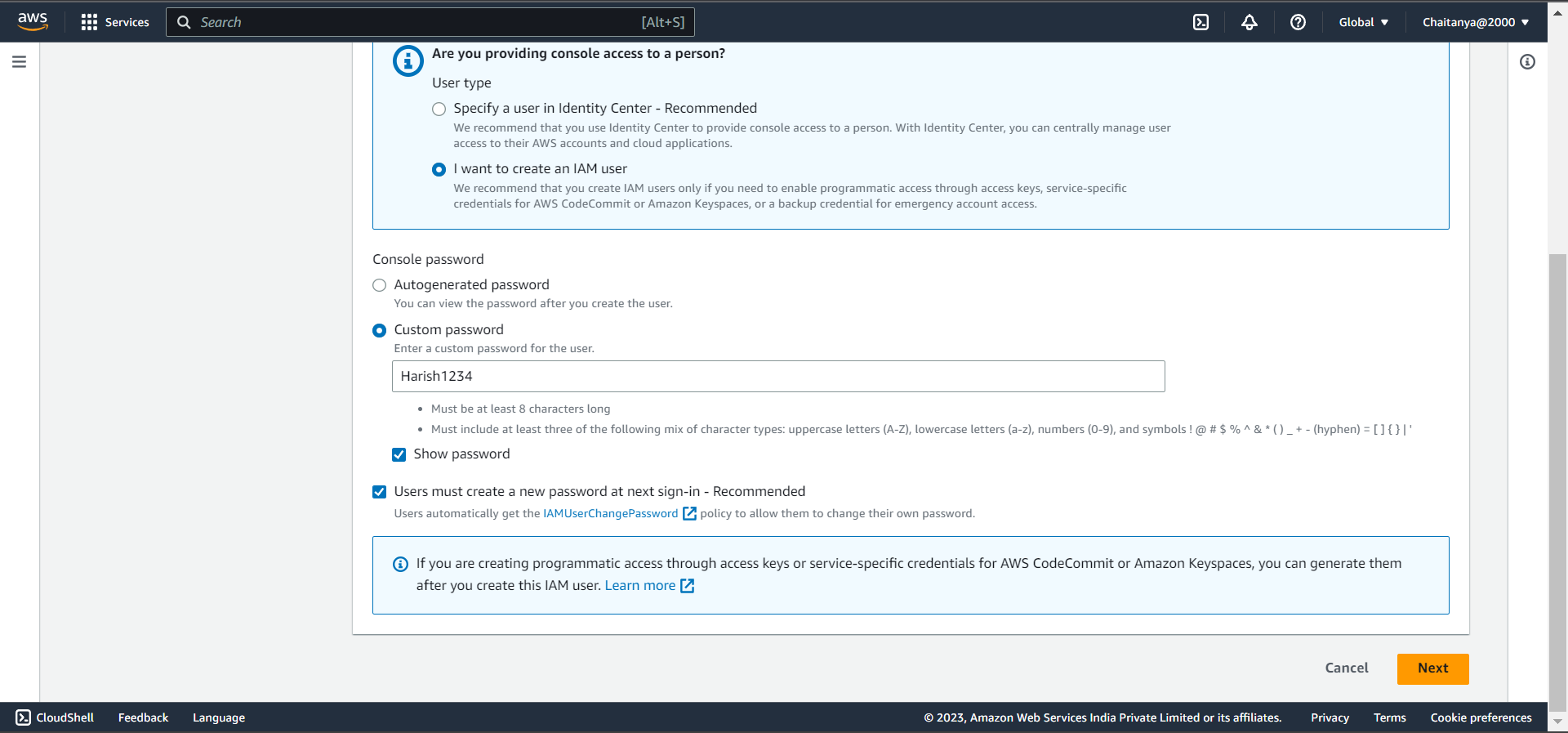
For programmatical access we need to provide access keys and secret keys so that he/ she can access the services programmatically.

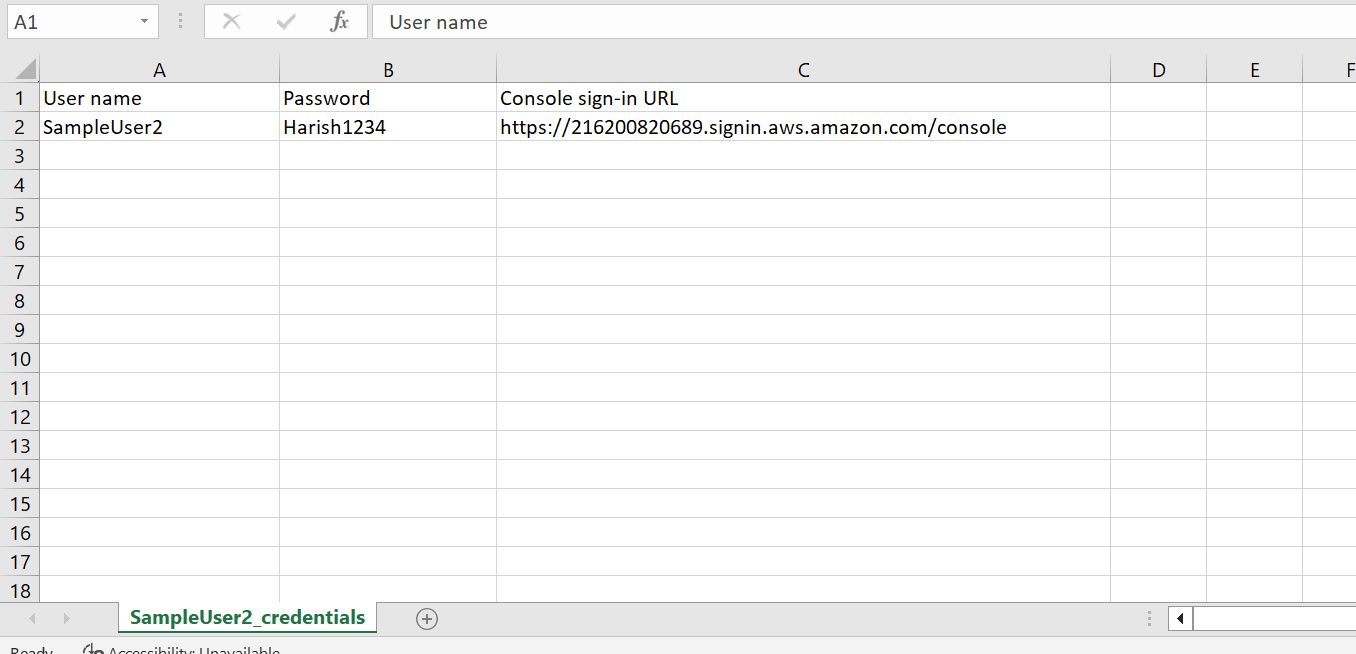
1. We can add permissions. We have 3 options for adding permissions.
2. Add a user to group
3. Copy permissions
4. Attach policies directly
5. Select a permission and click on add permission.
6. Now click on create user.
7. Click create an IAM user with some permissions
8. Click on Download.csv file which will download login credentials.

Now you can login using IAM User.

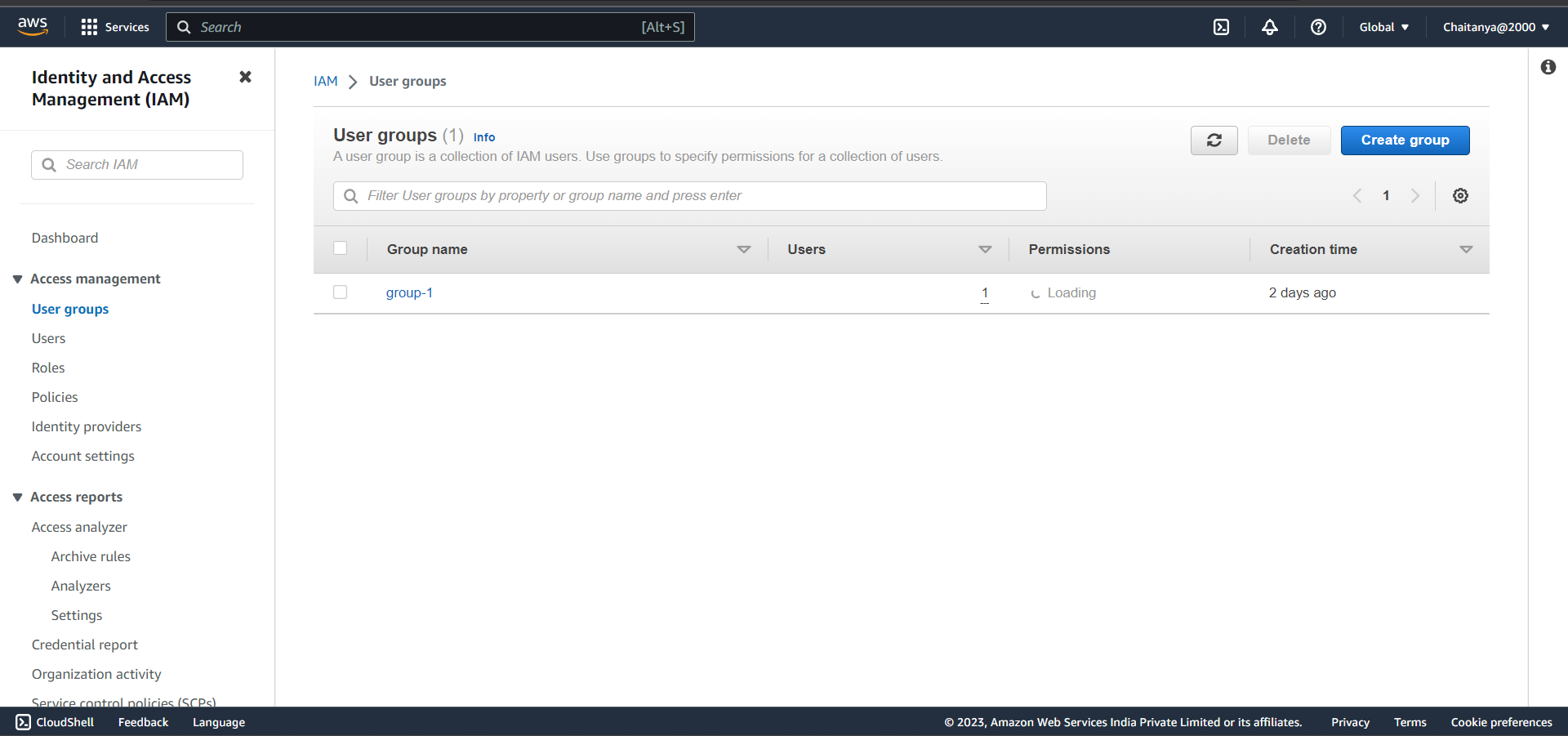
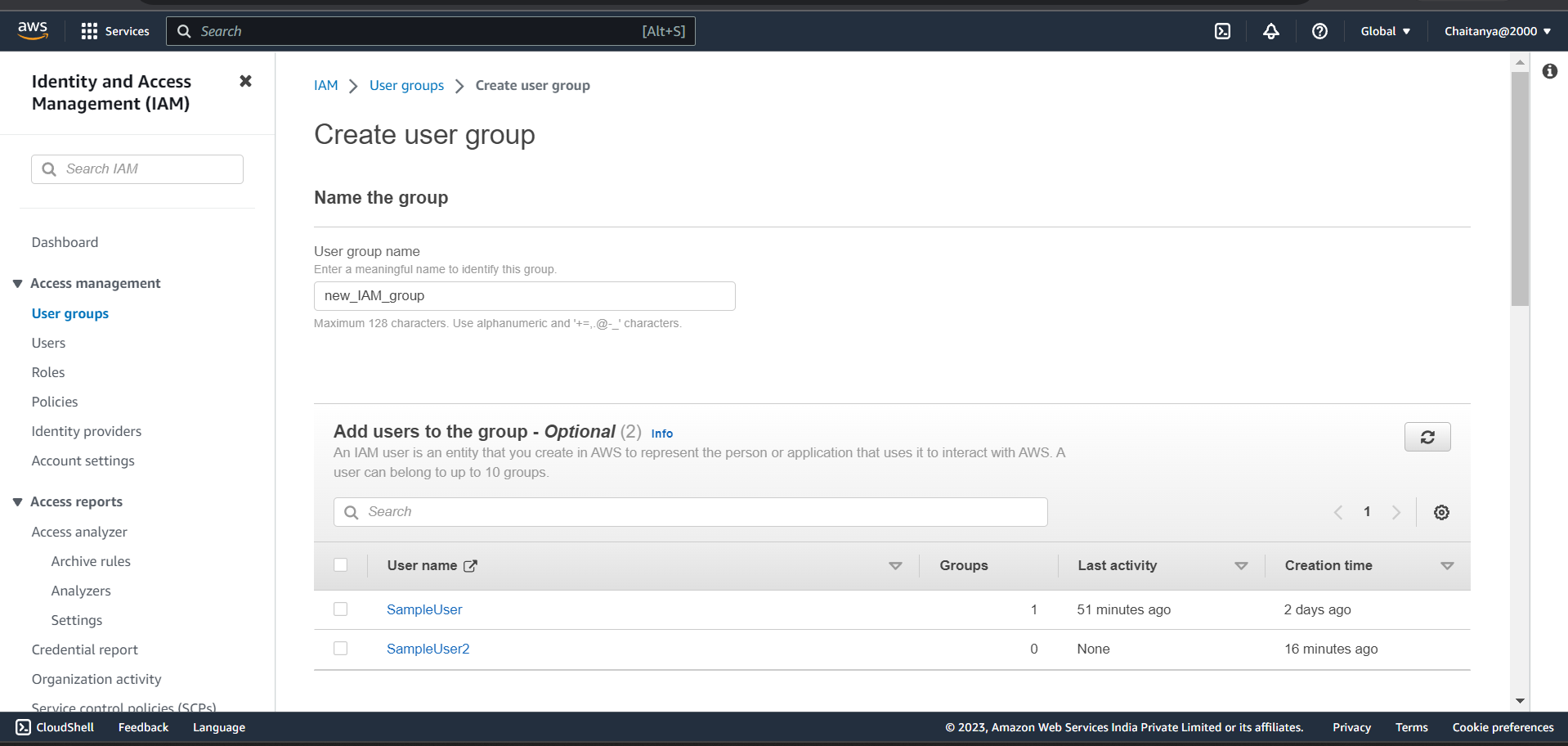
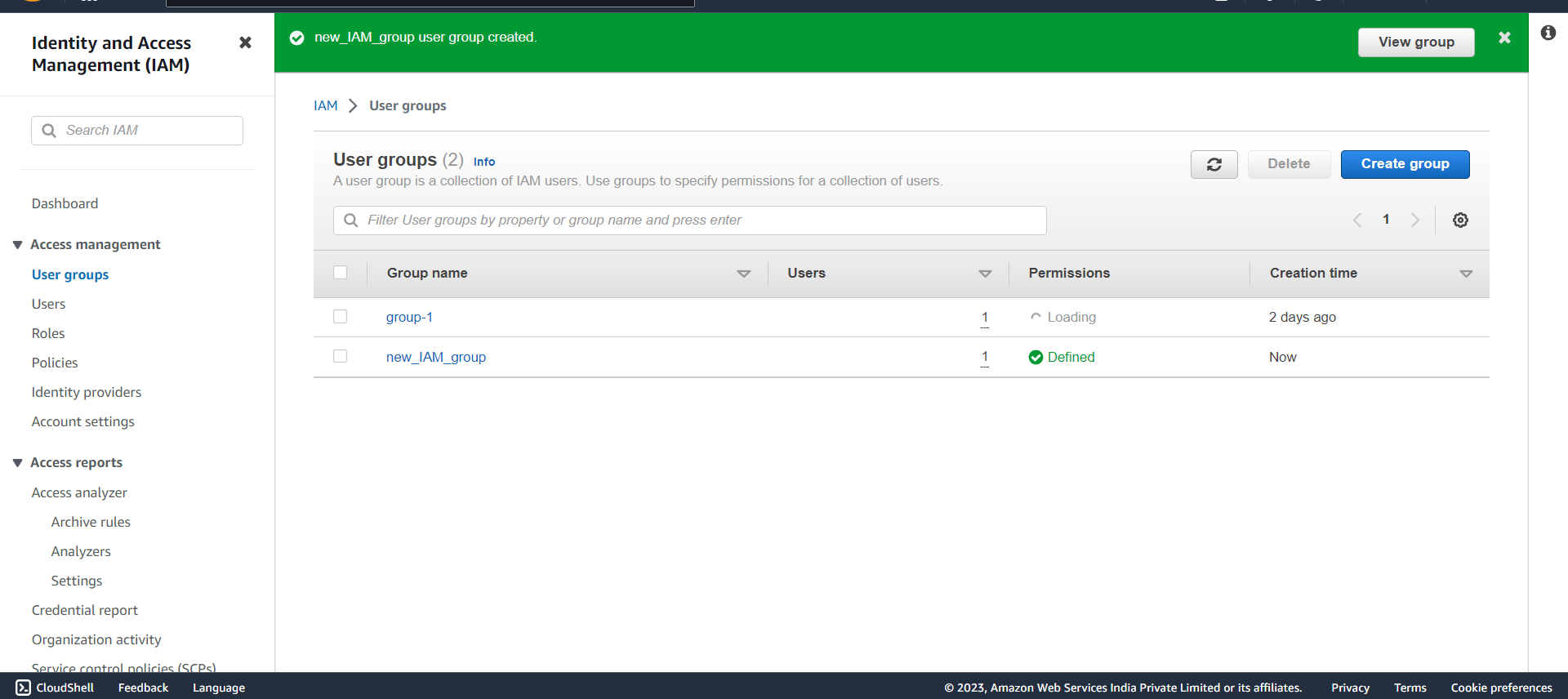








**Creating Groups:**

1. Navigate to IAM dashboard and select User groups from access management.
2. Enter group name
3. Select users to add
4. Attach policies
5. Click on create group
6. Now a new group is created.
7. 
8. 
9. 
10. 