## A REAL TIME PROJECT ON AWS(AMAZON WEB SERVICES)

## PROJECT 1 ON IAM(IDENTITY ACCESS MANAGEMENT)

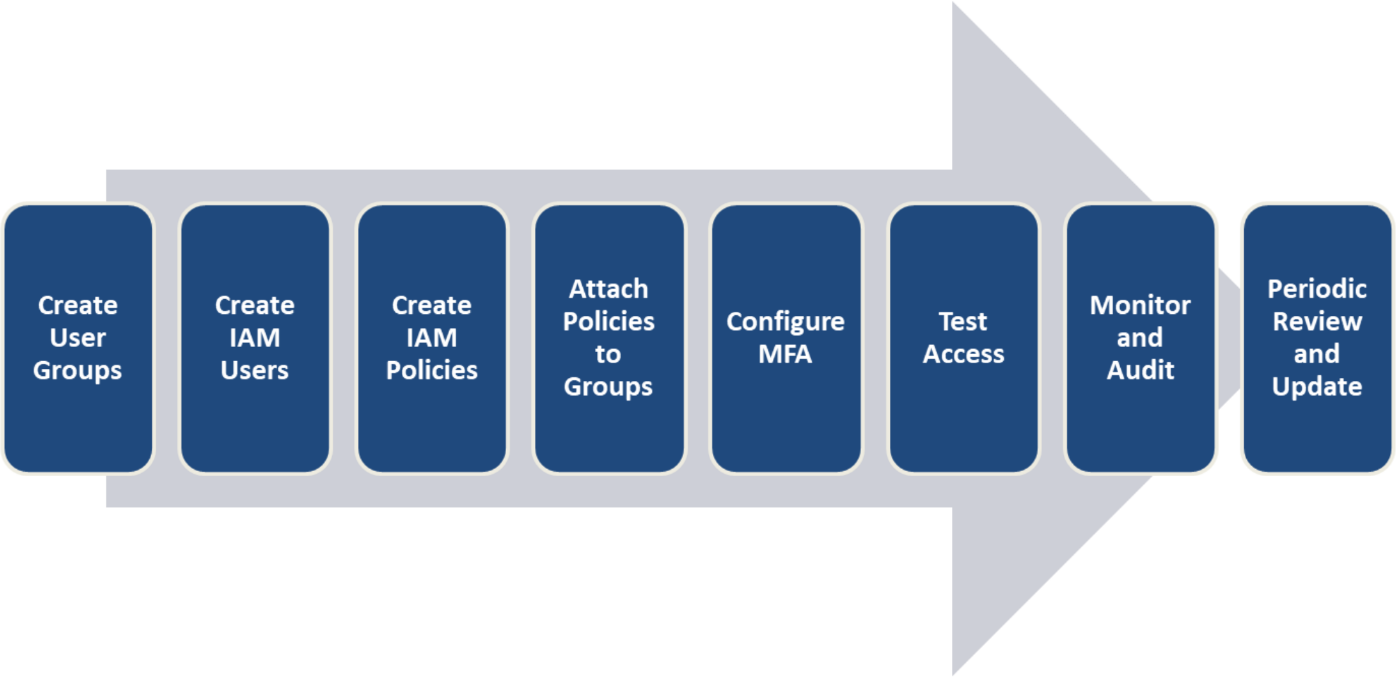
**A real-time project on AWS IAM (Identity and Access Management) involves setting up user accounts, roles, permissions, and policies to manage access to AWS resources.**

**In this example, let's create a project for a fictitious company called "TechCorp" that needs to manage AWS IAM for its developers, operations team, and third-party contractors.**

**Project Overview**:

**TechCorp has multiple AWS services running, including EC2 instances, S3 buckets, and RDS databases. They want to ensure that:**

1. **Developers have access to EC2 instances and RDS for application development.**
2. **Operations teams can manage all AWS resources.**
3. **Third-party contractors have limited access to specific S3 buckets.**

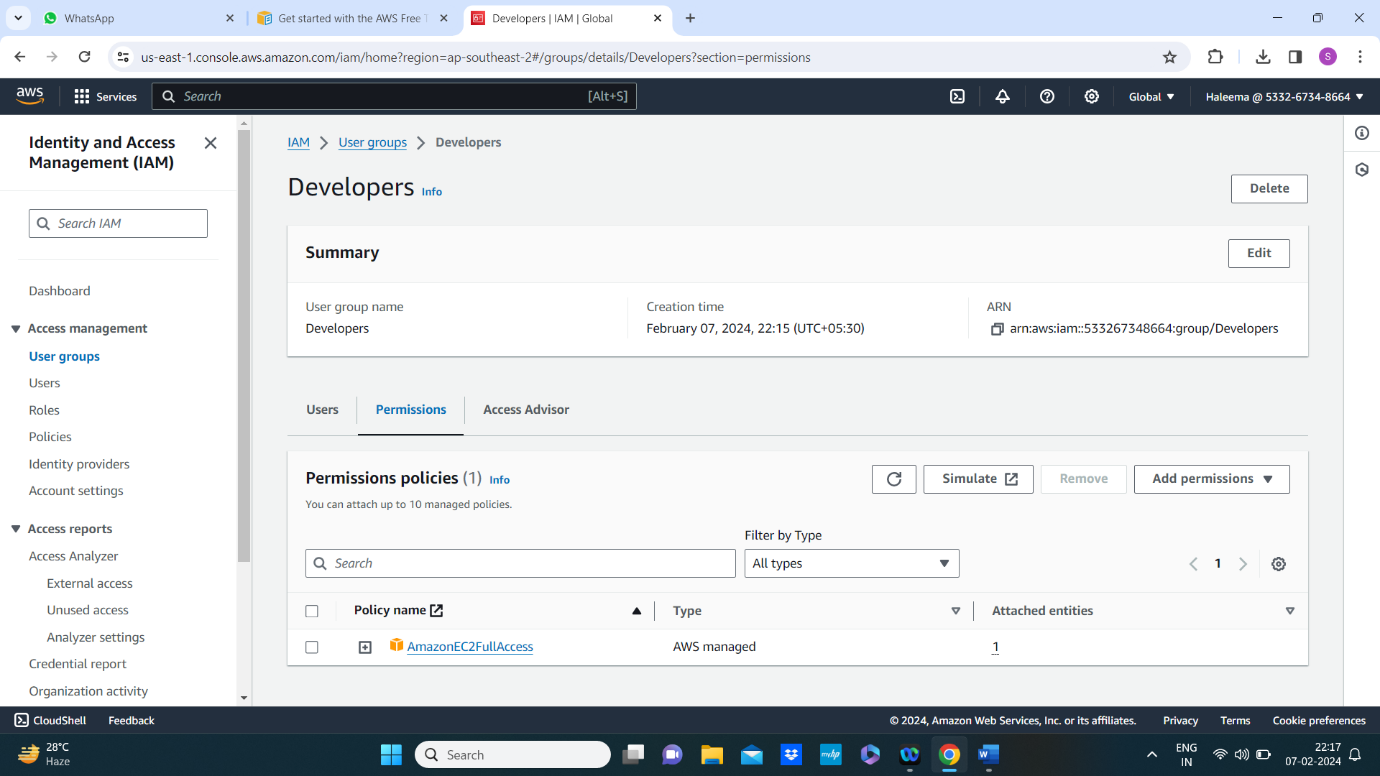


**Step 1: Create User Group**

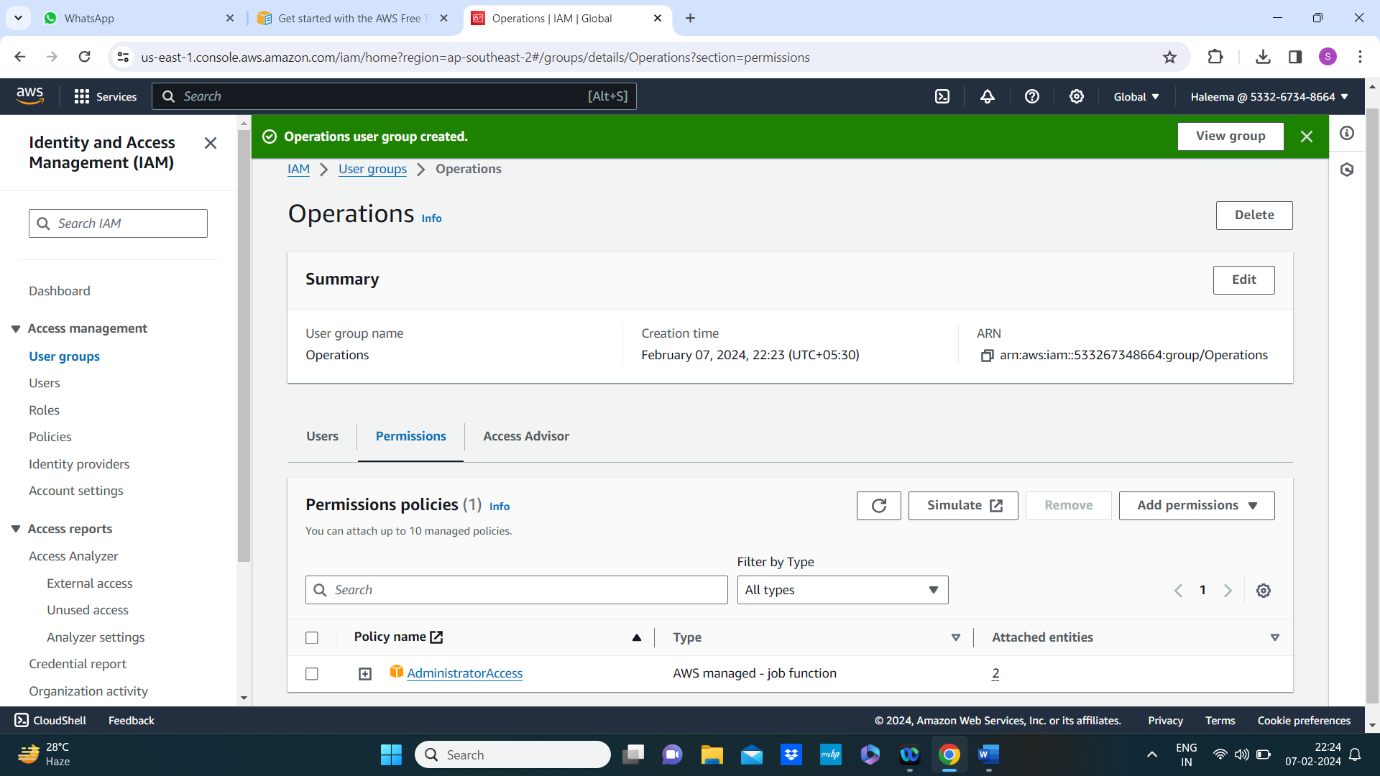
* **Going to create three IAM Groups**

1. **Developers Group**
2. **Operations Group**
3. **Contractors Group**

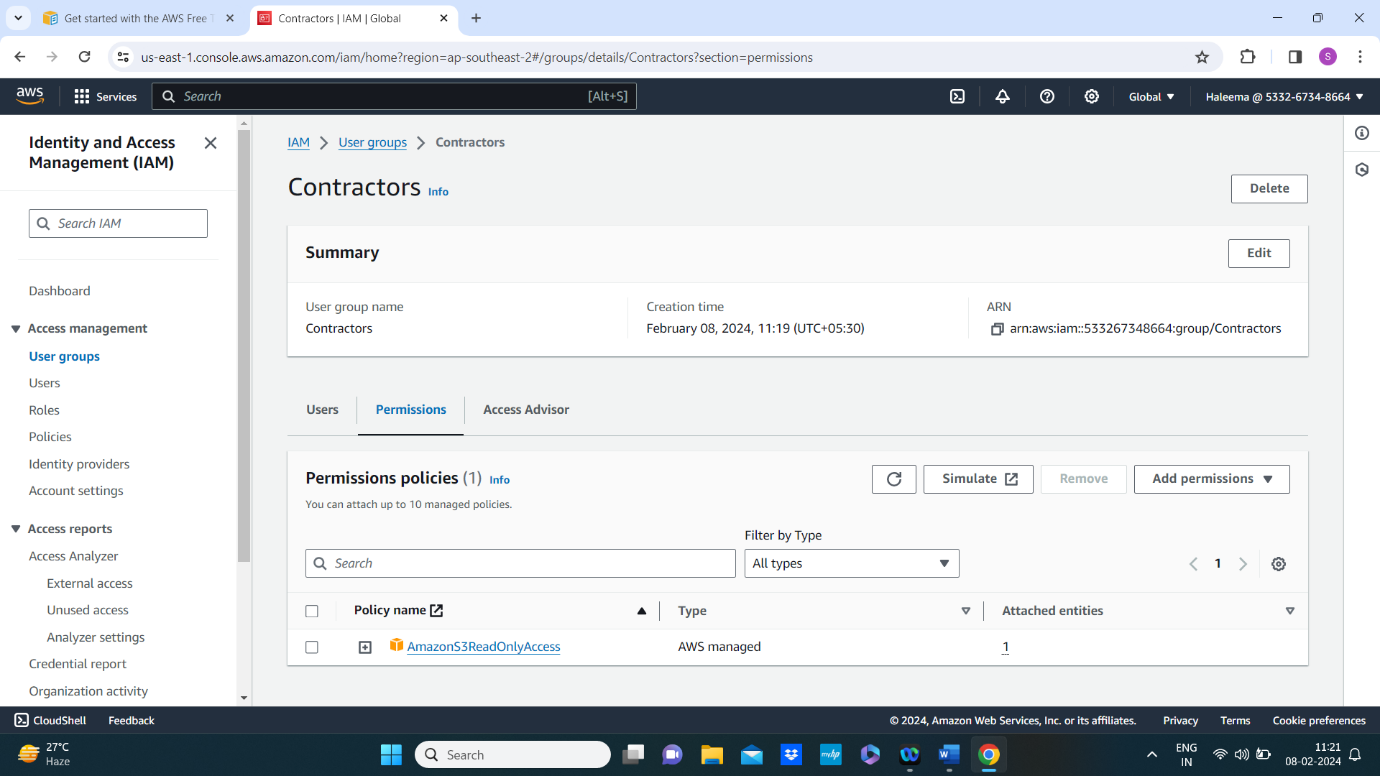
**Developer Group having EC2 Access is Created**

****

**Operations Group Having Full access is Created**

****

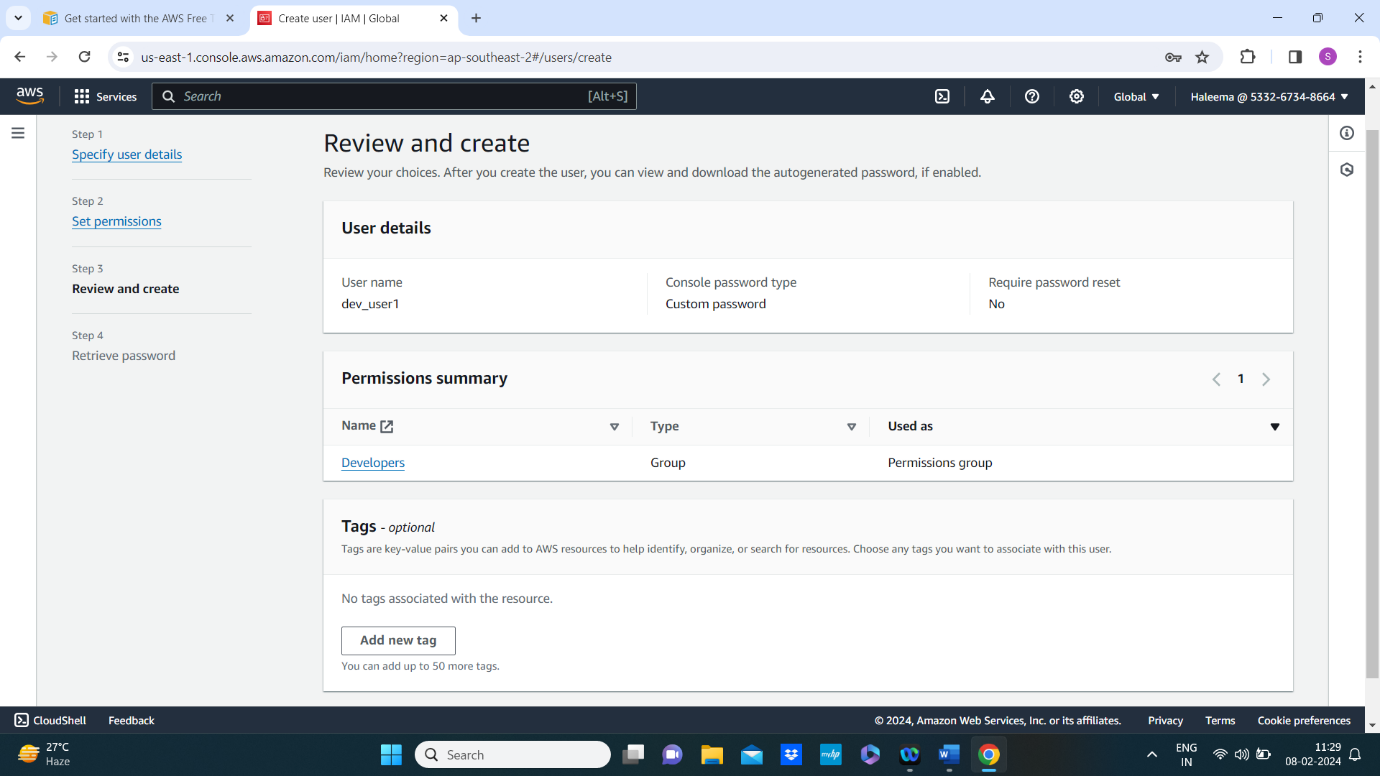
**Contractors Group having limited s3 access is created**

****

**Step 2 : Create IAM Users**

* **For developers group the users are dev\_user1 and dev\_user2**

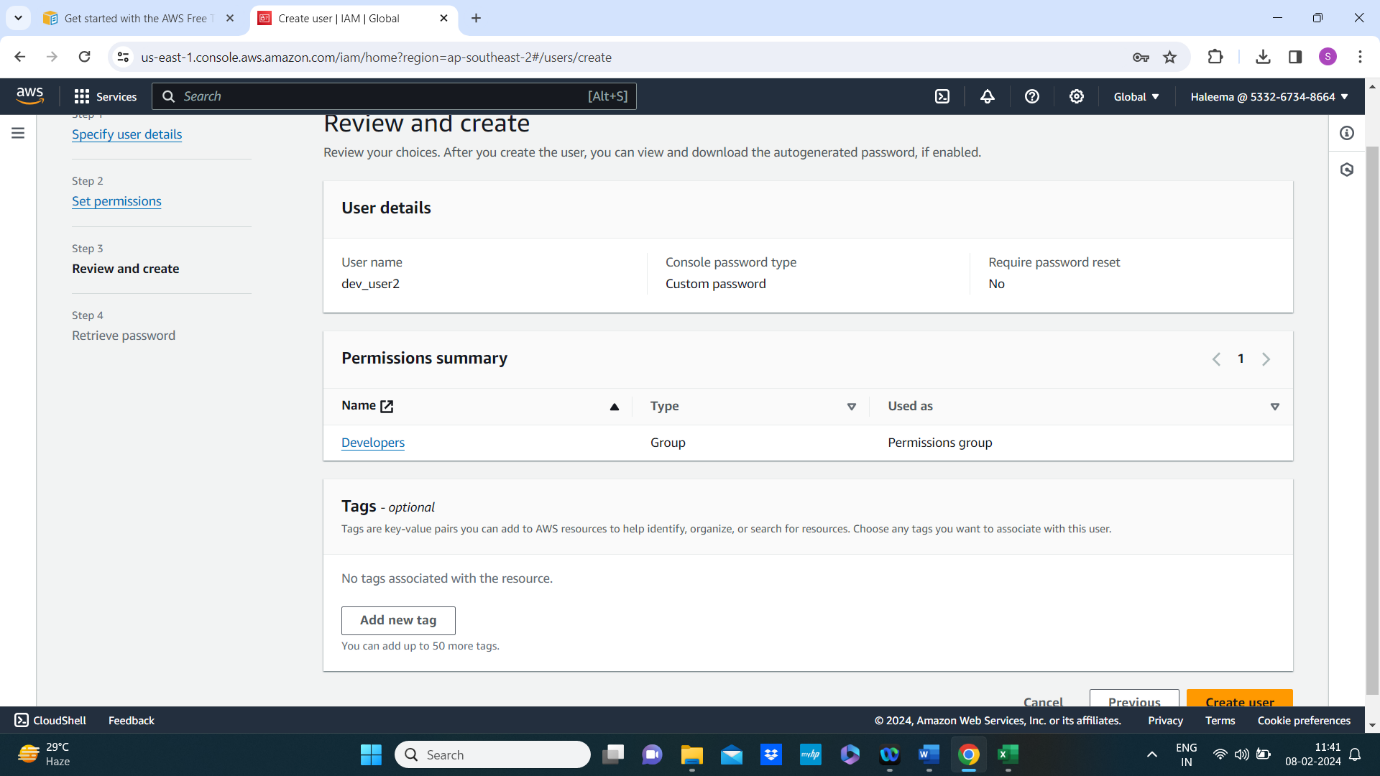
**dev\_user1 is Created. And assign it to developer group.**

****

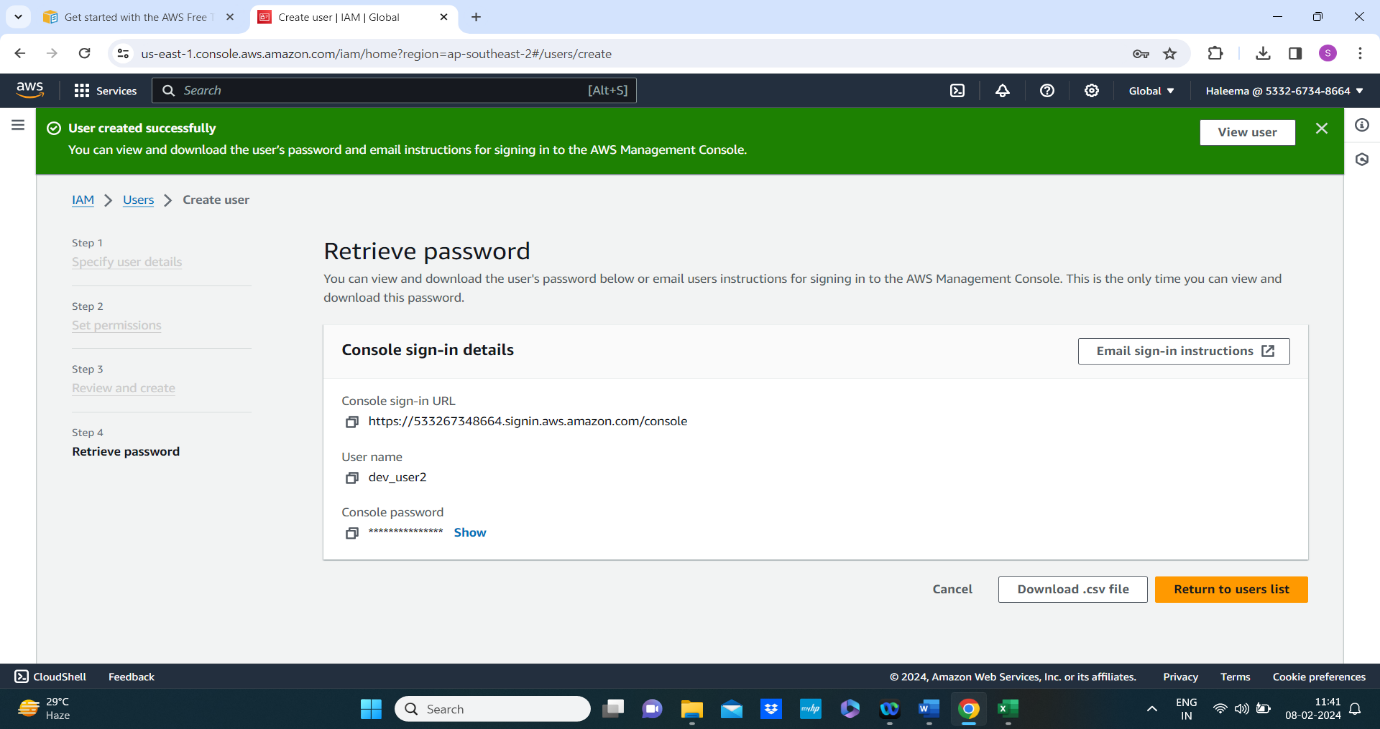
* **And the csv file is downloaded with user credentials**

|  |  |  |
| --- | --- | --- |
| **User name** | **Password** | **Console sign-in URL** |
| **dev\_user1** | **dev1@123** | **https://533267348664.signin.aws.amazon.com/console** |

* **Now creating dev\_user2 and assigning it to developers group**

****

* **And the csv file is downloaded with user credentials**

****

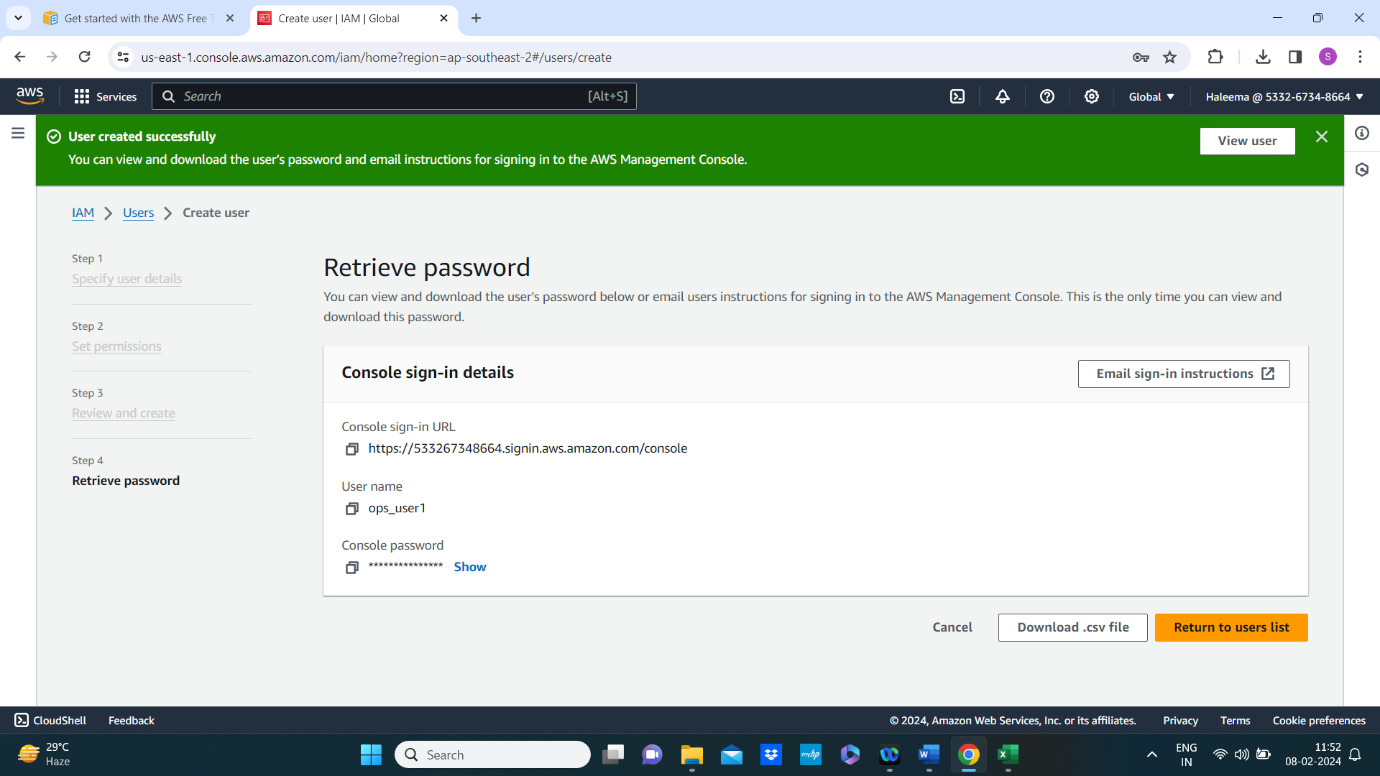
|  |  |  |
| --- | --- | --- |
| **User name** | **Password** | **Console sign-in URL** |
| **dev\_user2** | **dev2@123** | **https://533267348664.signin.aws.amazon.com/console** |

* **Creating ops\_user1 and ops\_user2 and assign it to operations group**

**Ops\_user1 is crated and assigned it to operations group**

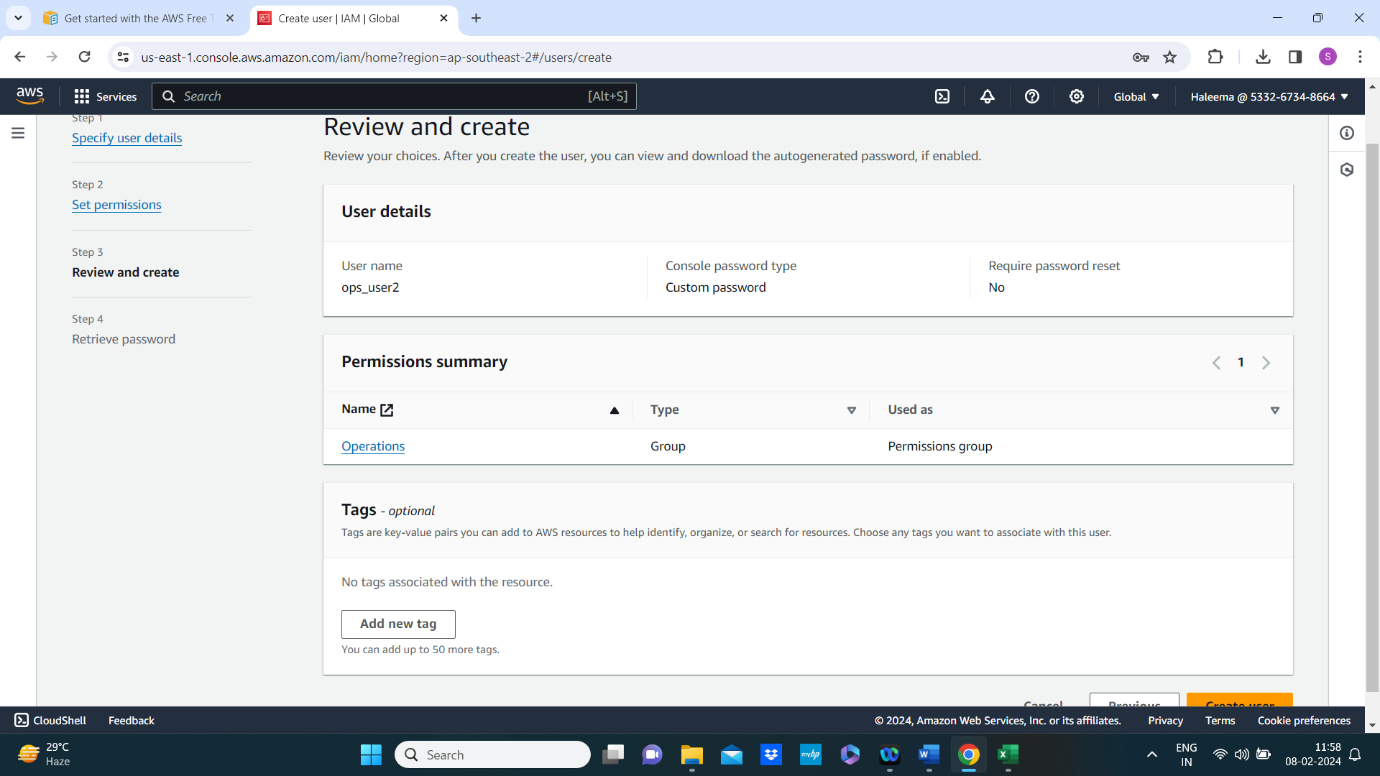
****

**And the csv file is downloaded**

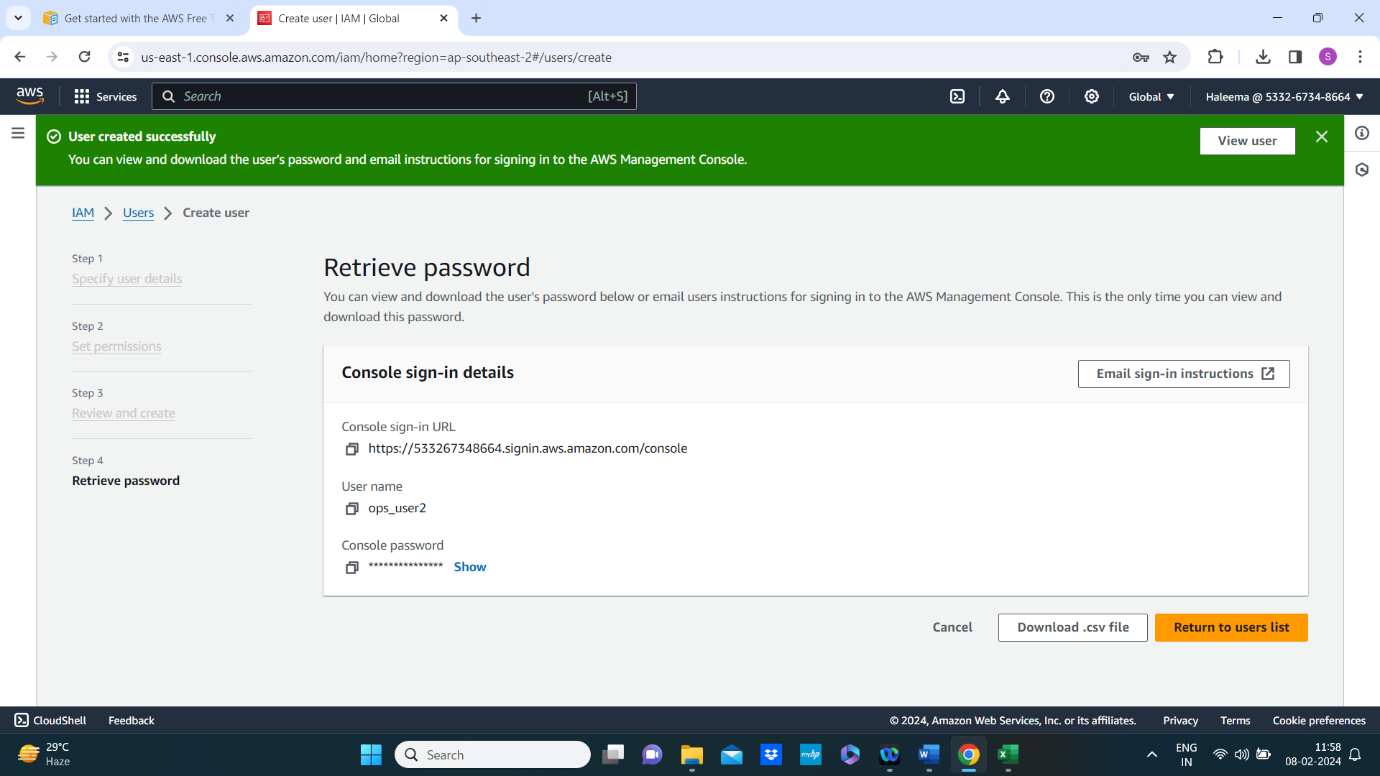
****

|  |  |  |
| --- | --- | --- |
| **User name** | **Password** | **Console sign-in URL** |
| **ops\_user1** | **ops1@123** | **https://533267348664.signin.aws.amazon.com/console** |

**Now creating ops\_user2 and assign it to operations group**

****

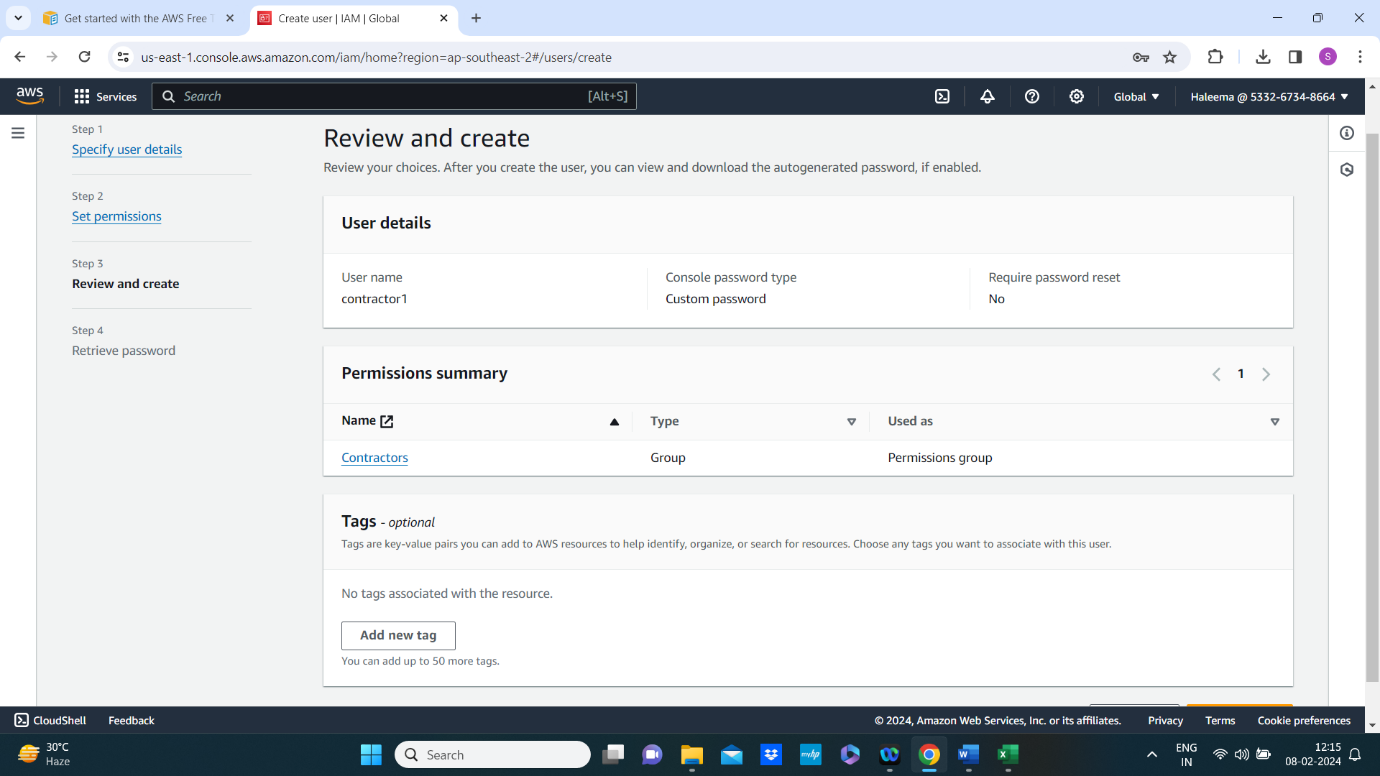
**And the csv file is downloaded**

****

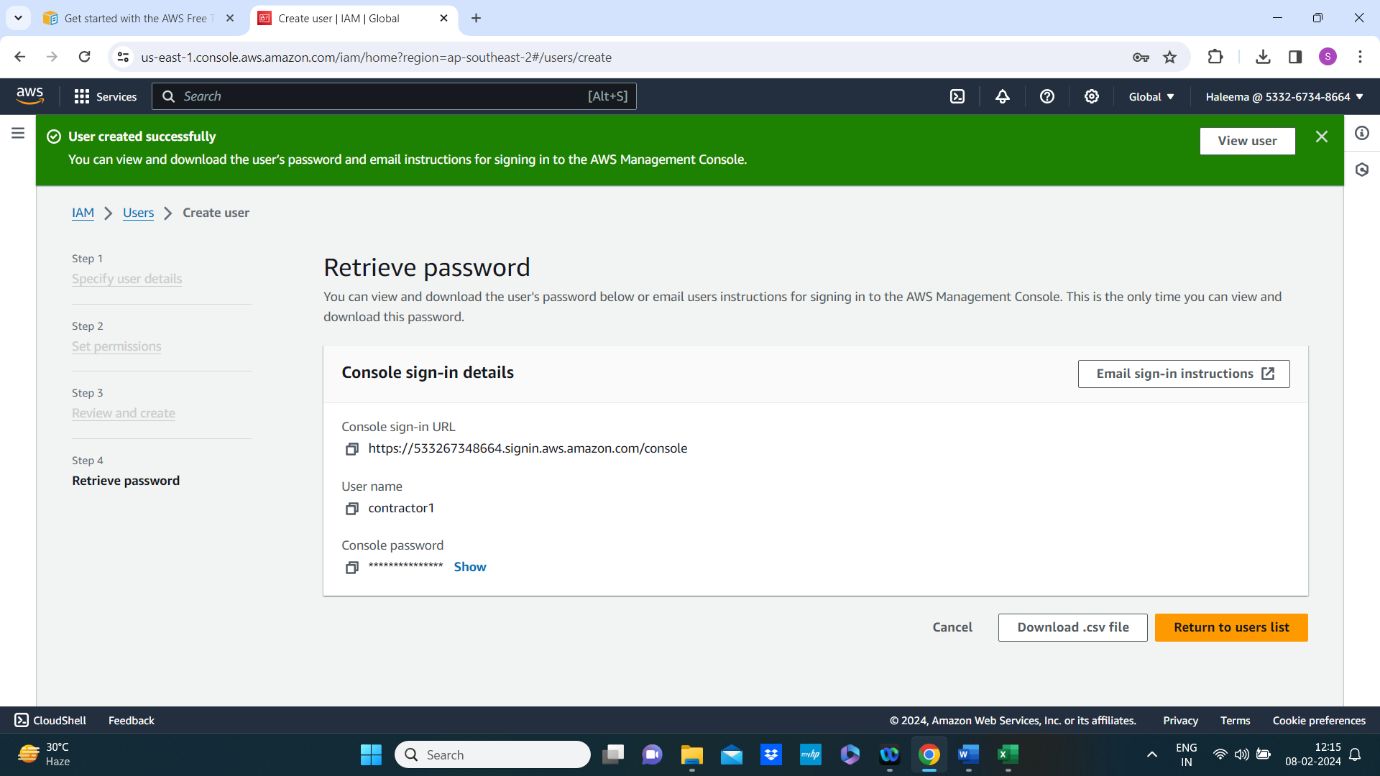
**User name Password Console sign-in URL**

**ops\_user2 ops2@123** [**https://533267348664.signin.aws.amazon.com/console**](https://533267348664.signin.aws.amazon.com/console)

* **Creating two users contractor1 and contractor2 and assign it to constructor group**

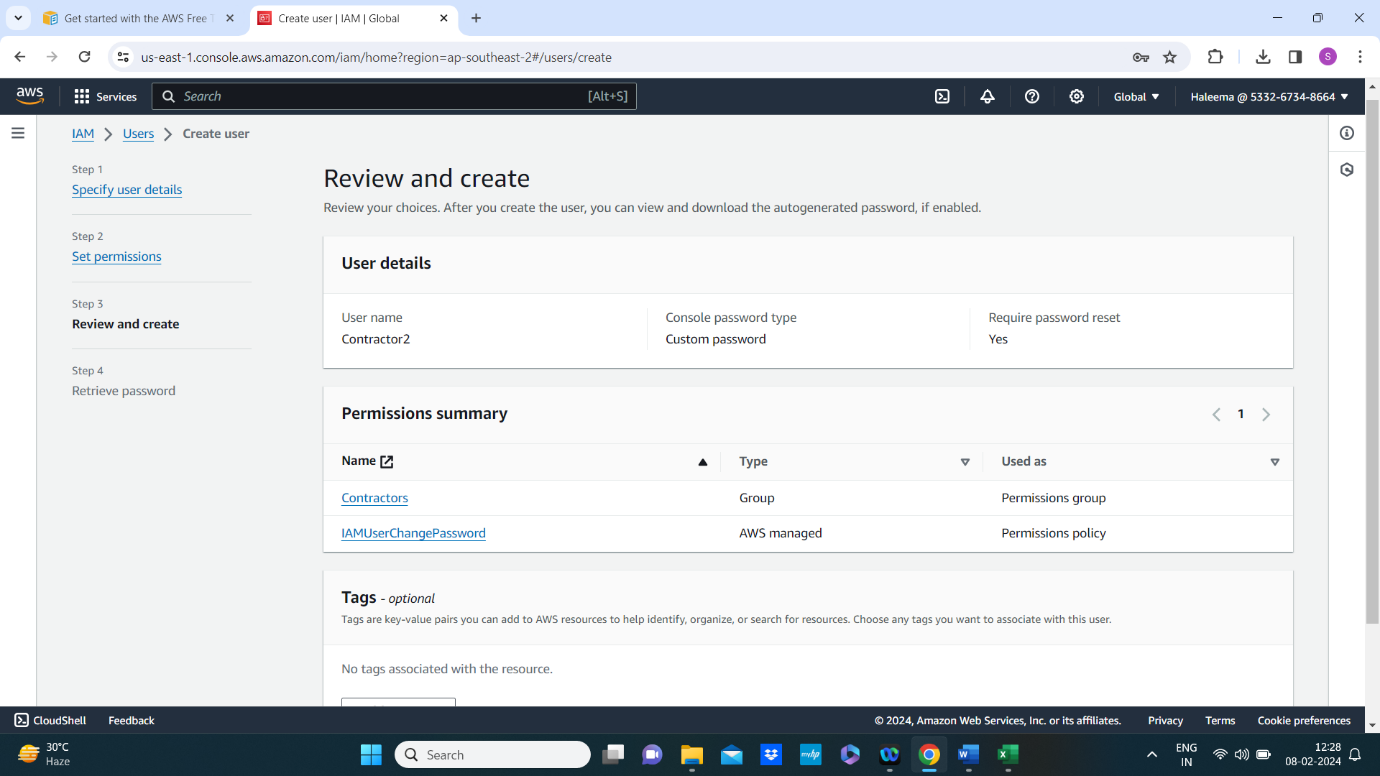
****

**And the csv file is downloaded**

****

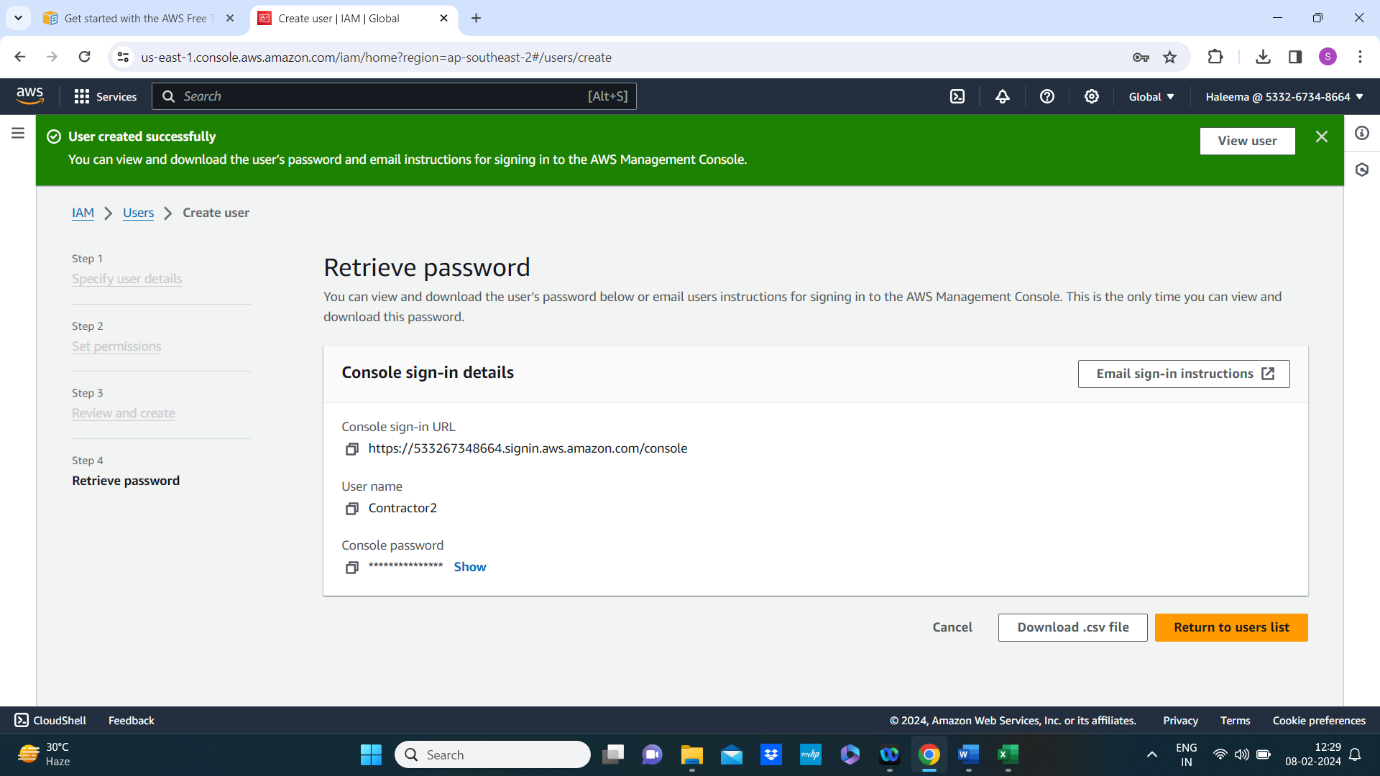
|  |  |  |
| --- | --- | --- |
| **User name**  **Contractor1** | **Password**  **Con1@123** | **Console sign-in URL**  [**https://533267348664.signin.aws.amazon.com/console**](https://533267348664.signin.aws.amazon.com/console) |

**Now create user contractor2 and assign it to Contractors group**

****

**This time user needs to change the password at the time of sign in to account.**

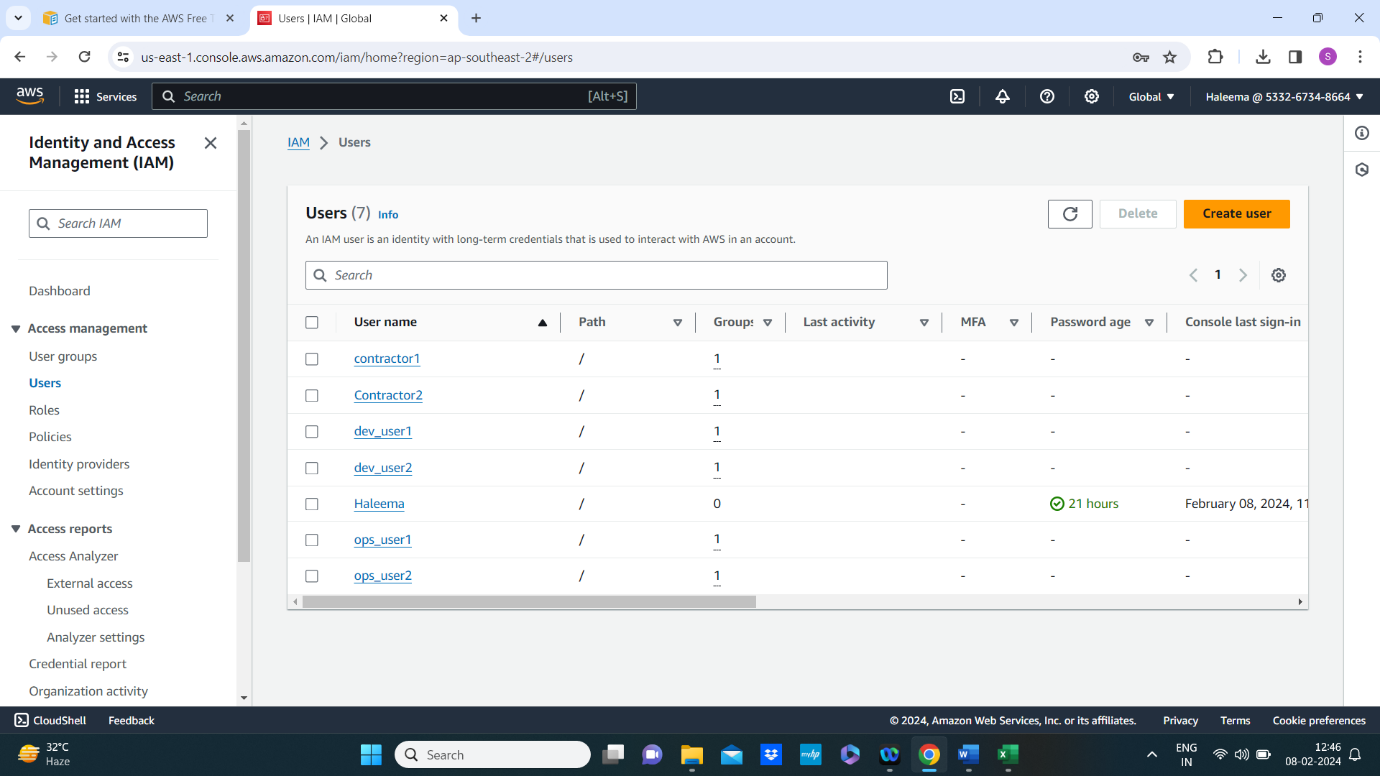
**And the csv file is also downloaded.**

****

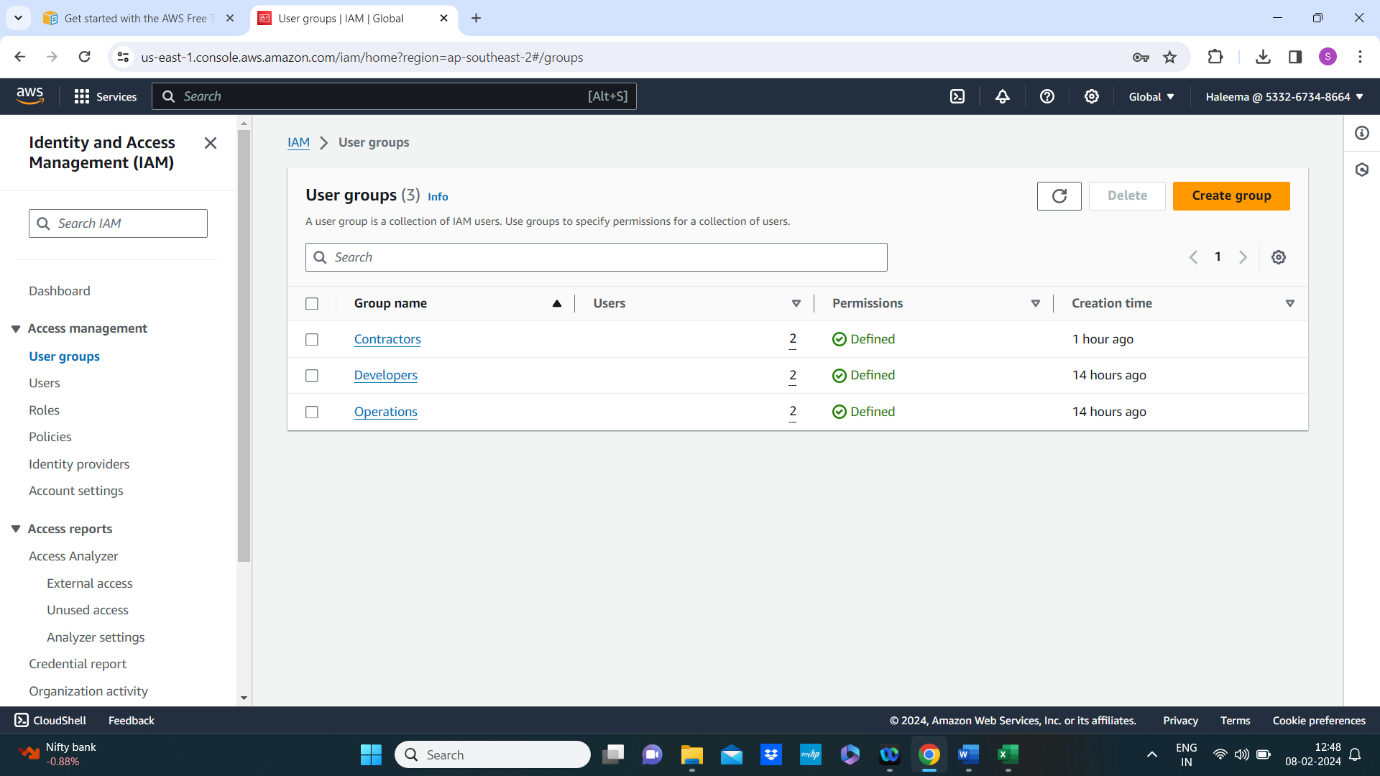
**User name Password Console sign-in URL**

**Contractor2 con2@123** [**https://533267348664.signin.aws.amazon.com/console**](https://533267348664.signin.aws.amazon.com/console)

**Here All the users are created**

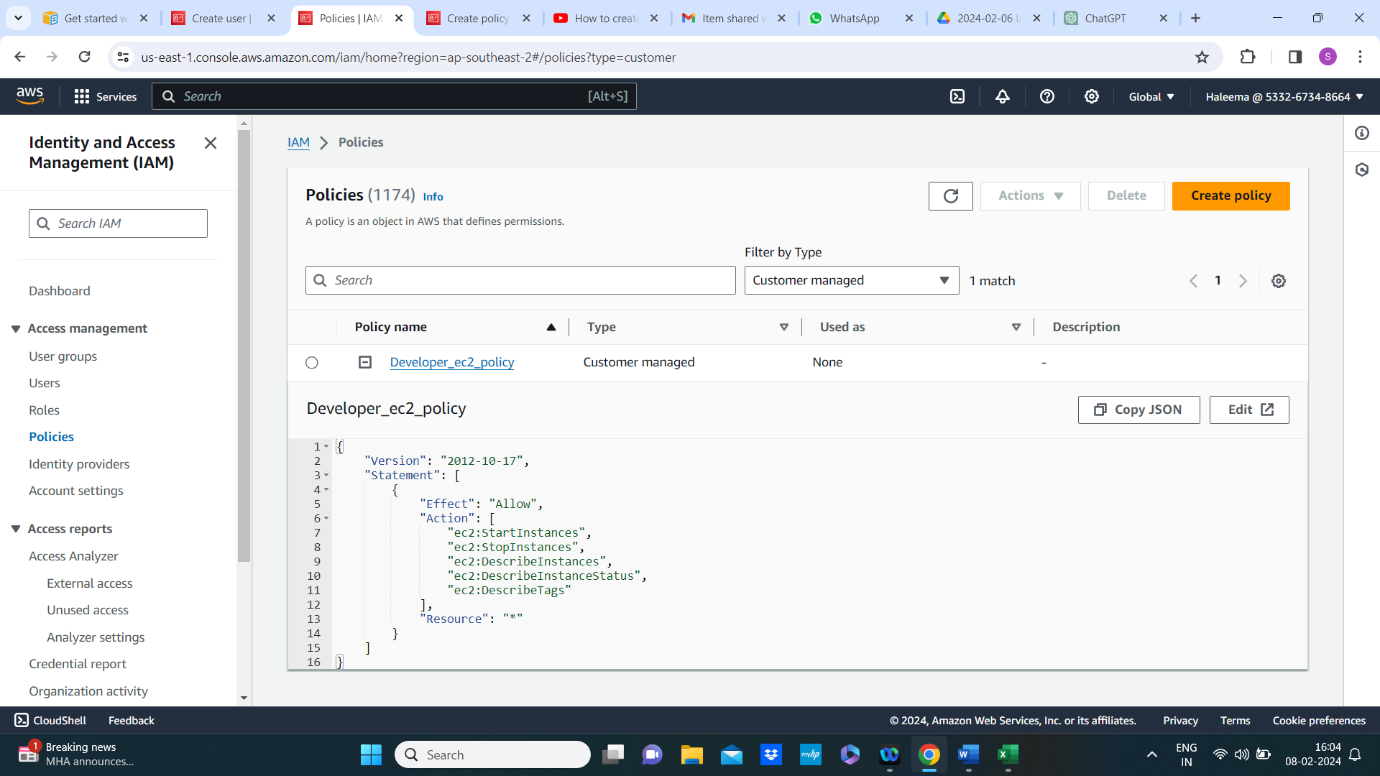
****

**Till now all the groups are created**

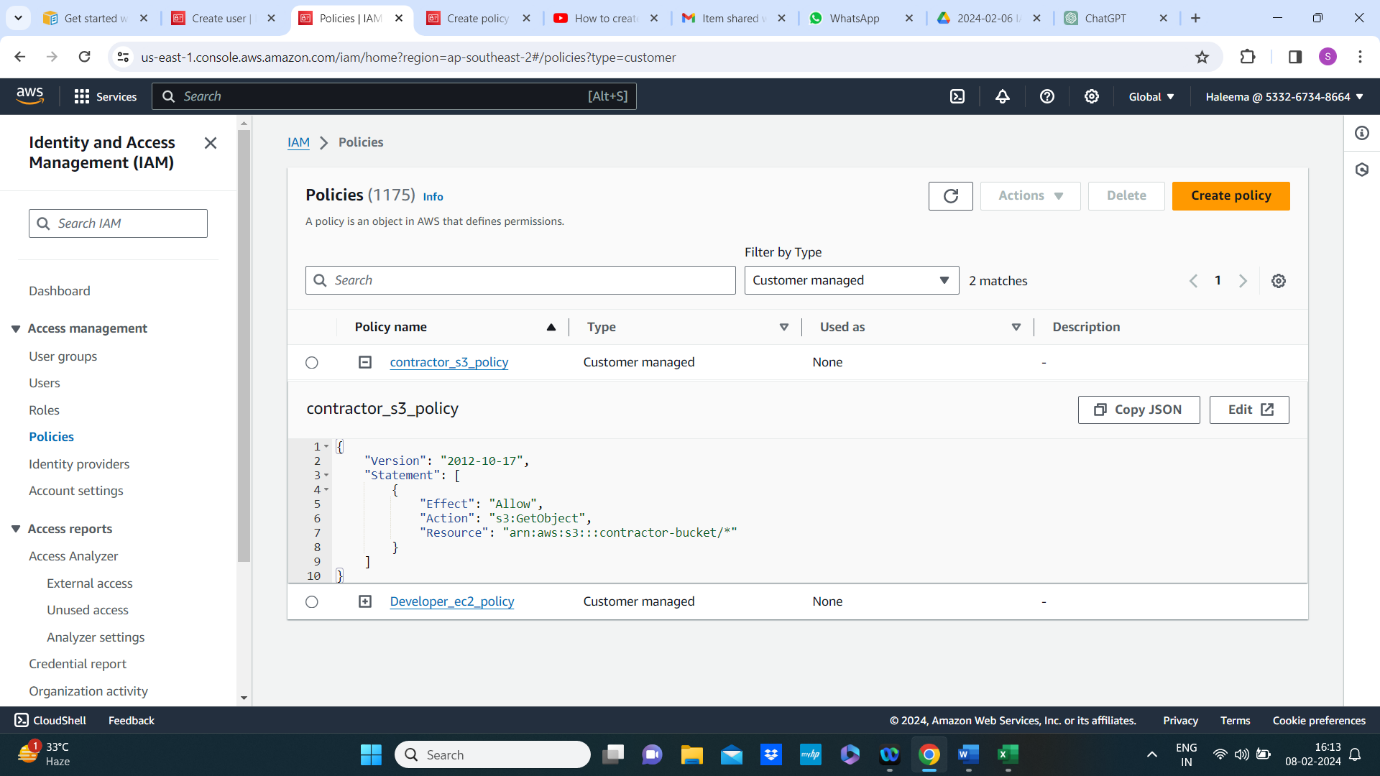
****

**Step 3: Create IAM Policies**

* **Creating a policy for developers by allowing ec2:StartInstances, ec2:StopInstances, ec2:DescribeInstances, and other relevant EC2 actions.**

****

* **Creating s3 policy for contractor to allow read access only to specific s3 buckets, e.g.,s3: GetObject for contractor-bucket.**

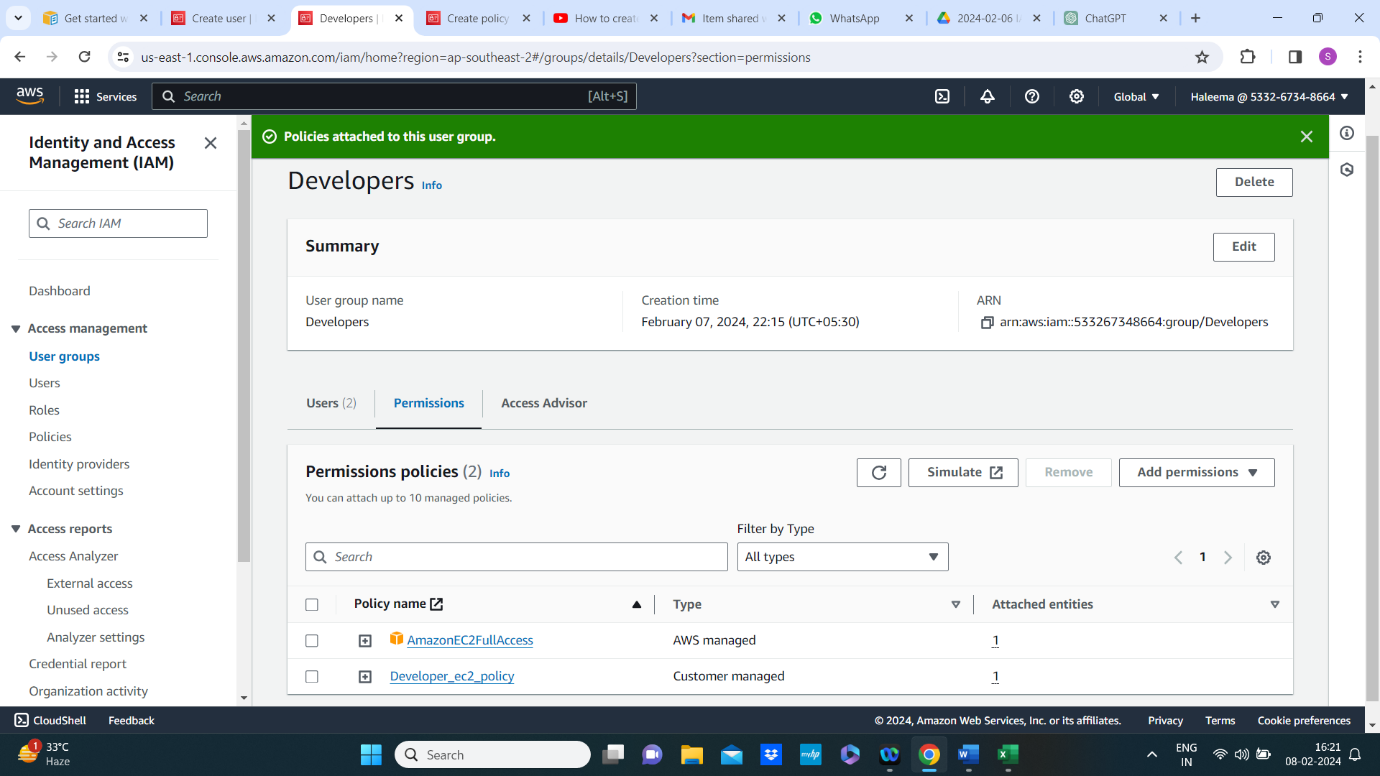
****

* **Policy for developers and contractors are created.**

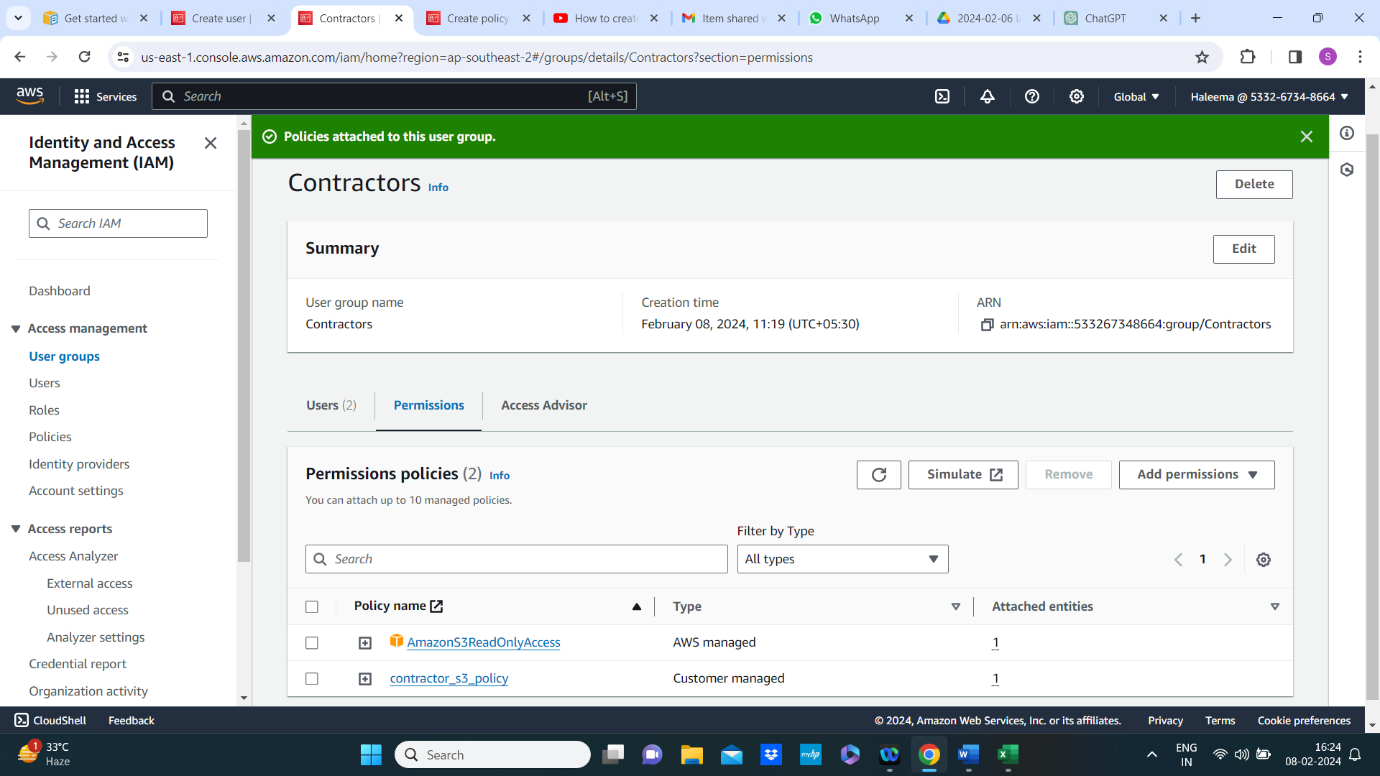
****

**Step 6: Attach policies to Groups**

* **Attach developer policy to developers group**

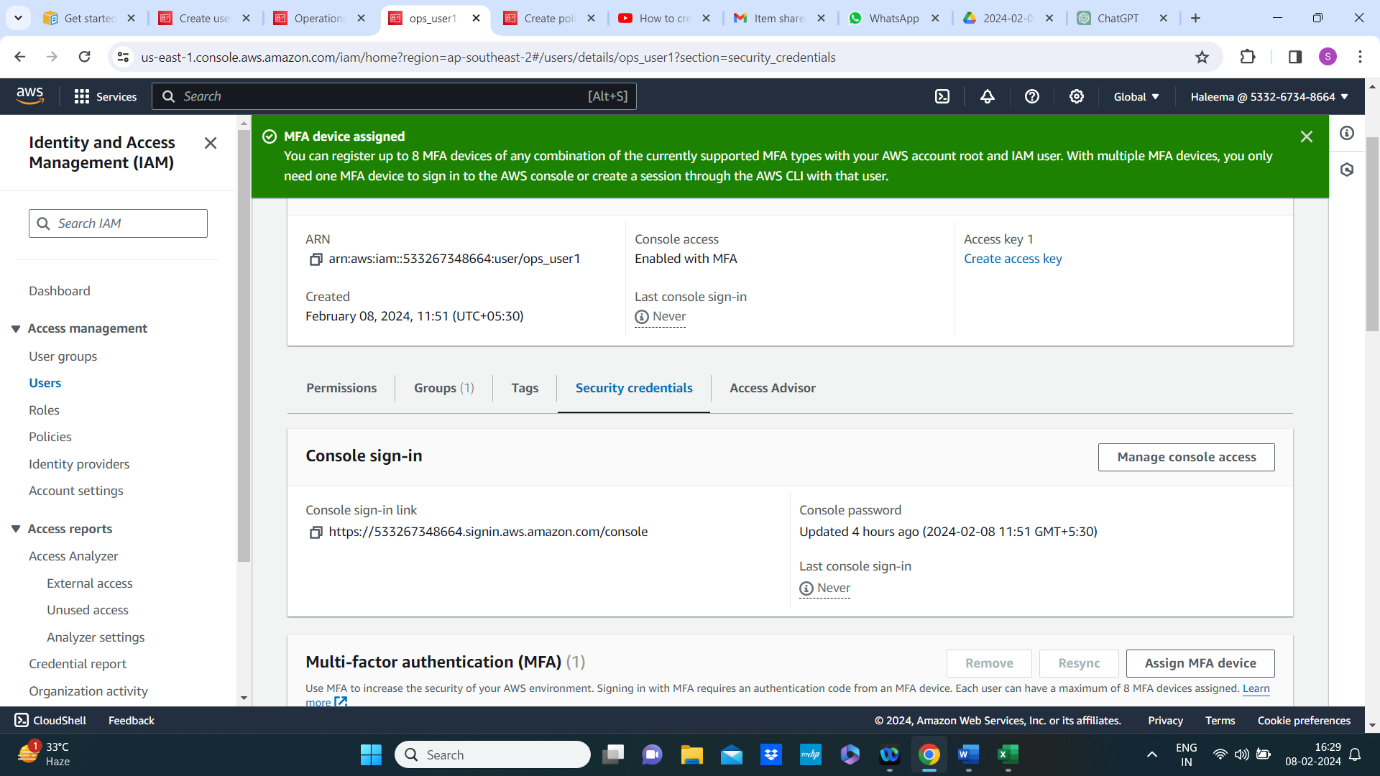
****

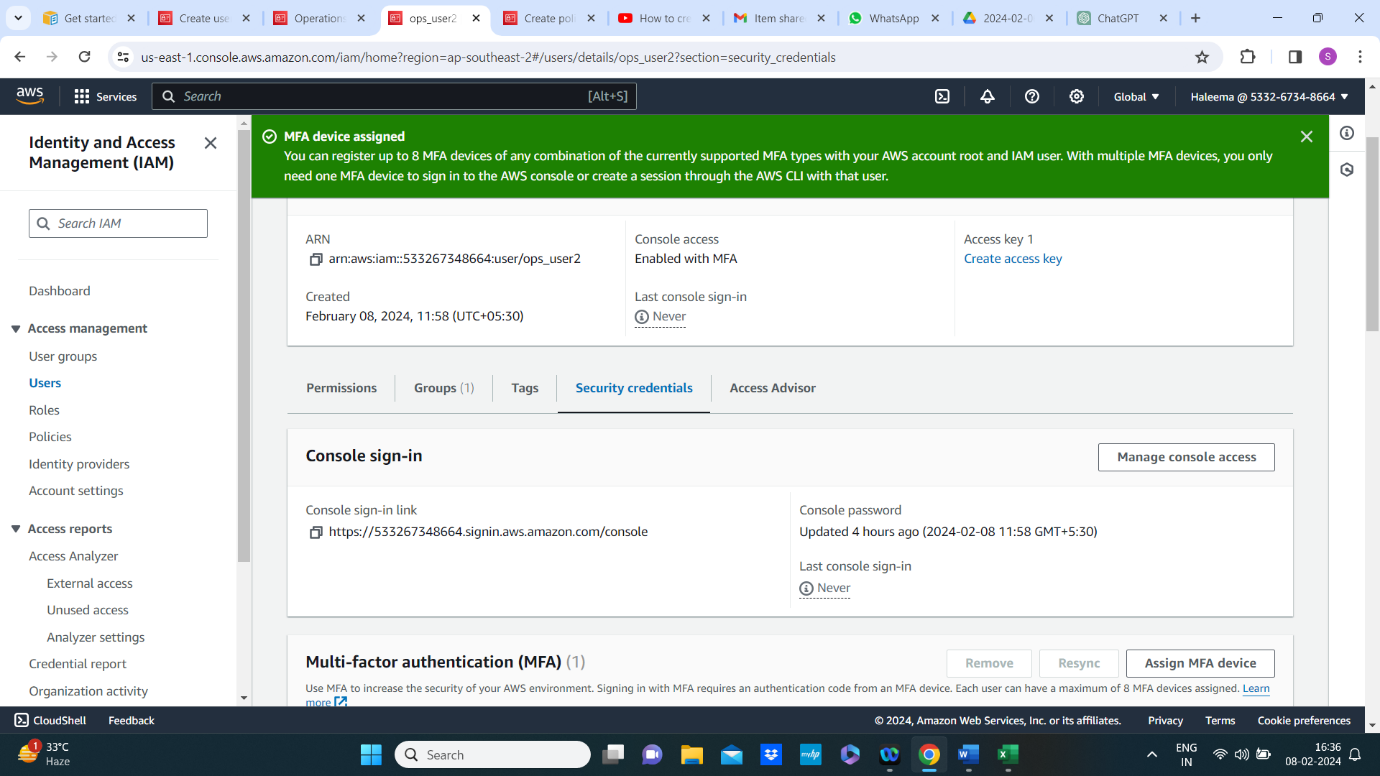
* **Attach contractor policy to contractors group**

****

**Step 5: Configure MFA(Multi-Factor Authentication)**

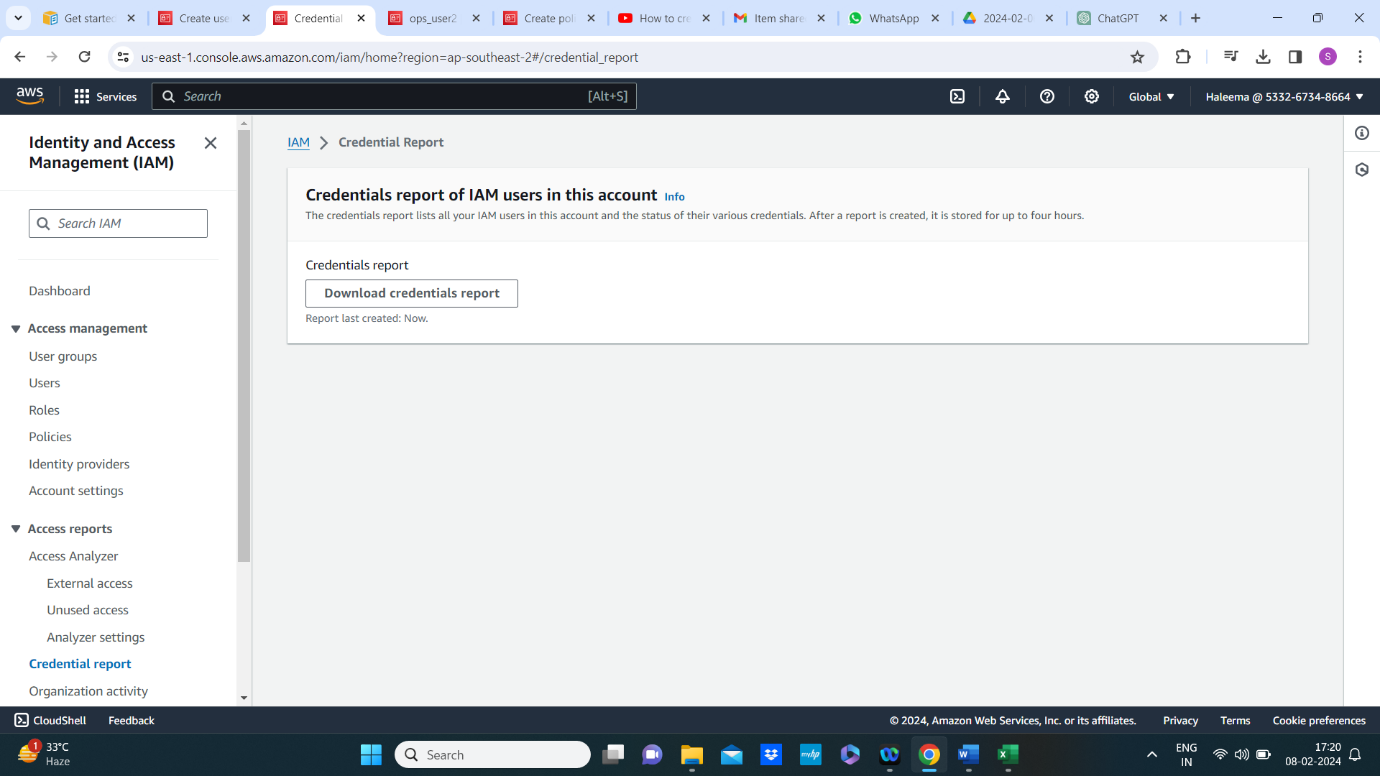
* **Enable MFA for users especially for users in the operations group**

****



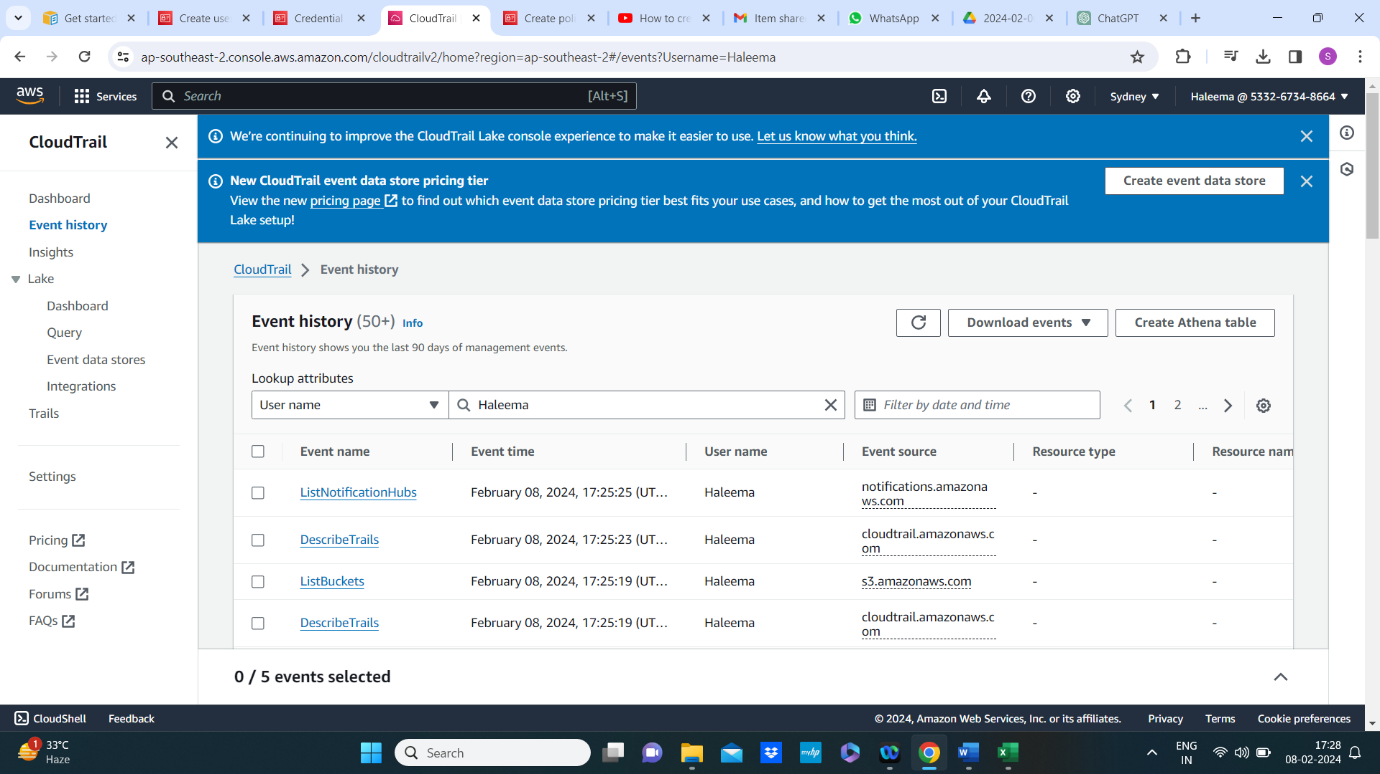
Step 6: test access

* Having the developers try to start and stop EC2 instances
* Contractors should test accessing the specified s3 bucket.
* Operations team members should be able to manage all AWS resources



Step7: Monitor and Audit

* Implement AWS cloudwatch logs and AWS CloudTraill for monitoring and auditing access to resources
* Regularly review logs and access reports to ensure security and compliance.



Step 8: Periodic review and update

* Periodically reviewing IAM Permissions to ensure they align with the organization’s needs.
* Remove or modify access for users who no longer require it.