A MINI PROJECT ON

IAM SECURITY INFRASTRUCTURE

BY

Sayali Parhar

17 TH MARCH 2025

Abstract

This Project provides a secure IAM setup using AWS Identity and Access Management (IAM). The organization consists of multiple teams, each with different levels of access requirements. The goal is to implement a well-structured IAM security model using Users, Groups, Roles, and Policies with proper access controls for EC2, S3, RDS, and IAM services

Contents

Chapter 1

INTRODUCTION

1.0 IAM

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. With IAM, you can manage permissions that control which AWS resources users can access. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources. IAM provides the infrastructure necessary to control authentication and authorization for your AWS accounts.It lets you create identities, define permissions, and control what actions users can perform

- IAM ROOT USER: When you first create an Amazon Web Services (AWS) account, the email address and password you provide are the credentials for your root user, which has access to all AWS services and resources in the account.
- IAM USERS: An IAM user is an entity that you create in your AWS account. The IAM user represents the human user or workload who uses the IAM user to interact with AWS resources. An IAM user consists of a name and credentials. An IAM user with administrator permissions is not the same thing as the AWS account root user.
- IAM GROUPS: An IAM user group is a collection of IAM users. User groups let you specify permissions for multiple users, which can make it easier to manage the permissions for those users. For example, you could have a user group called Administrators and give those user groups typical administrator permissions. Any user in that user group automatically has Admins group permissions.

- IAM ROLES: An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM role is similar to an IAM user in that it is an AWS identity with permission policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a role is intended to be assumed by anyone who needs it. In addition, a role does not have standard long-term credentials such as a password or access keys associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session .

- IAM POLICY: A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when an IAM principal (user or role) makes a request. The permissions in the policies determine whether the request is allowed or denied.

Chapter 2

IAM ARCHITECHTURE

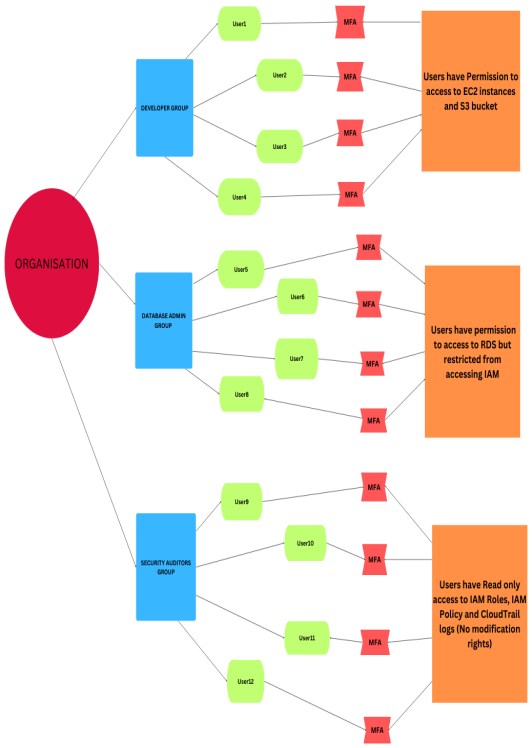

Figure 2.1: IAM Architechture

1.0 Working

As shown in the above Architechture, I have implementated an IAM Security Model for an Organization which divided into three teams which are Developers Team, Database Admins Team and Security Auditors Team. Each Team consists of four Users and each team have different access permission and according to that permissions they can able to access the AWS resources.

- Developers Team: Developers Team have access to EC2 Instances and S3 Buckets.

  In that they can control EC2 start instance,stop Instance and describe Instance attributes only.

  About S3 buckets they have read and write access so they can perform all read and write operations on s3 buckets.

- Database Admins Team: This Team have Full access to RDS and restrict users for accessing IAM resources.

  Full Access to RDS means database admin users can able to access all the RDS resources provided by AWS.

  Database Admin users dont have any access to IAM resources so, I completely restrict access of IAM for Database Admin group.

- Security Auditors Group: This Group can view IAM roles,policies and logs and they dont have any modification rights in AWS.

  View means they can only have read only access for IAM roles,policies and logs so,I apply read only access of IAM with specific List and read attributes by creating custom policy and for logs I apply cloudtrail read only policy.

2.0 Security

To enhance the security, I use MFA(Multifactor Authentication), MFA helps to protect AWS resources from unauthoarized users.
As per the need of an Organization I assign MFA login to each user, for that I use a phone as a virtual multi-factor authentication (MFA) device. whenever user try to login he will get a standards-based TOTP (time-based one-time password) which is six-digit long, after entering correct TOTP user can successfully login to the AWS console and access the asign aws resources.

Chapter 3

TEST CASES

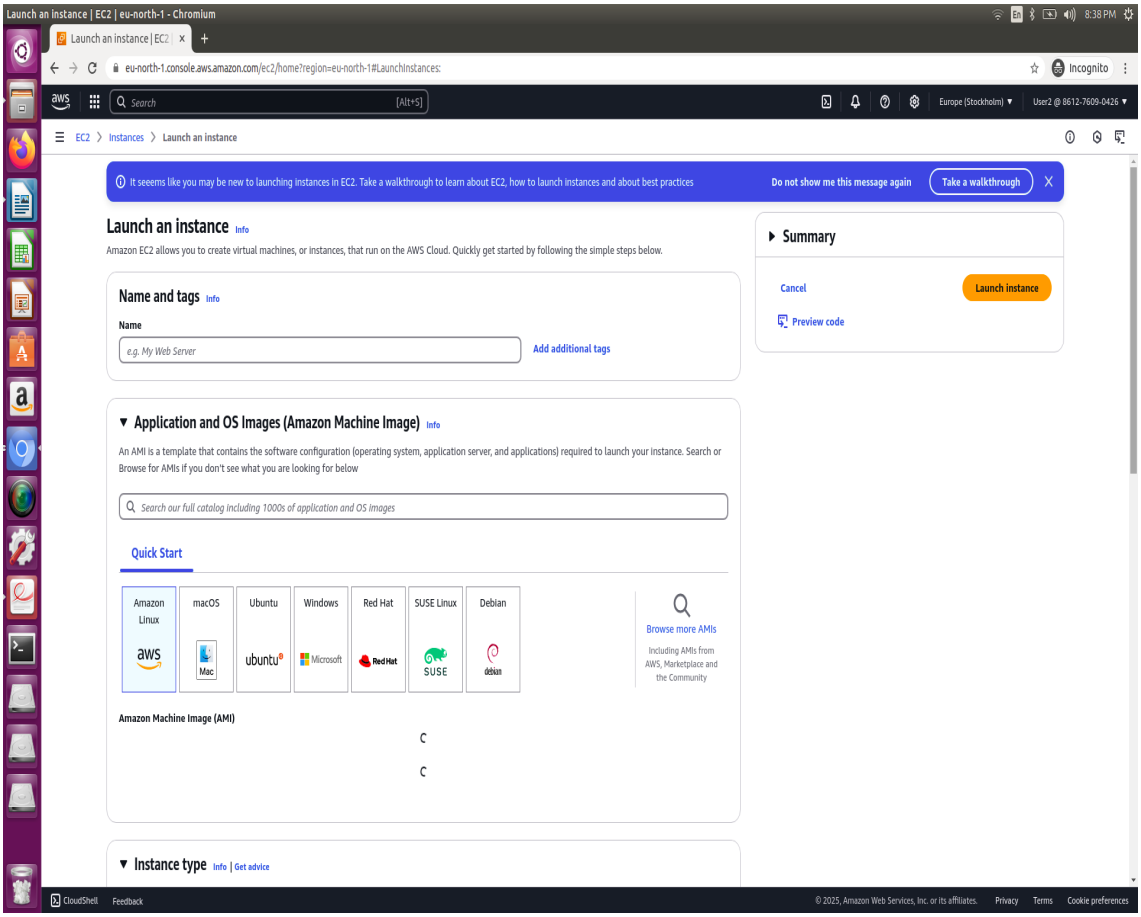

Figure 3.1: EC2instance

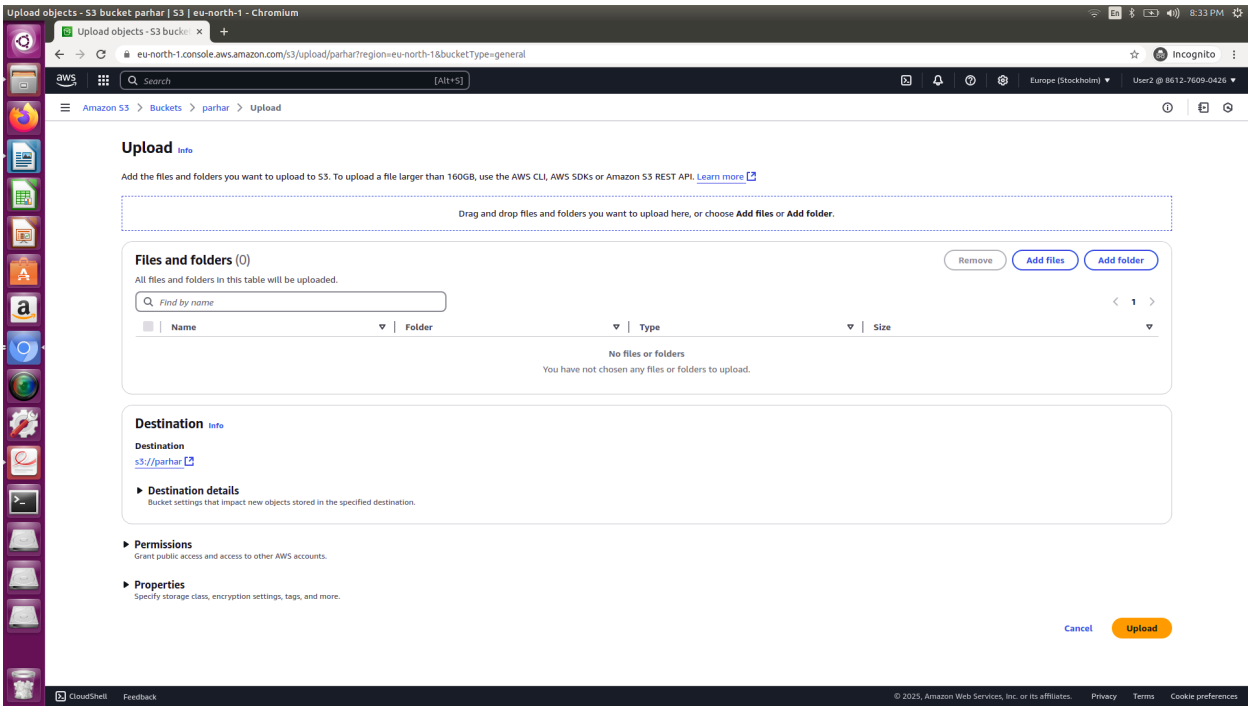

Figure 3.2: S3 write access through uploading option

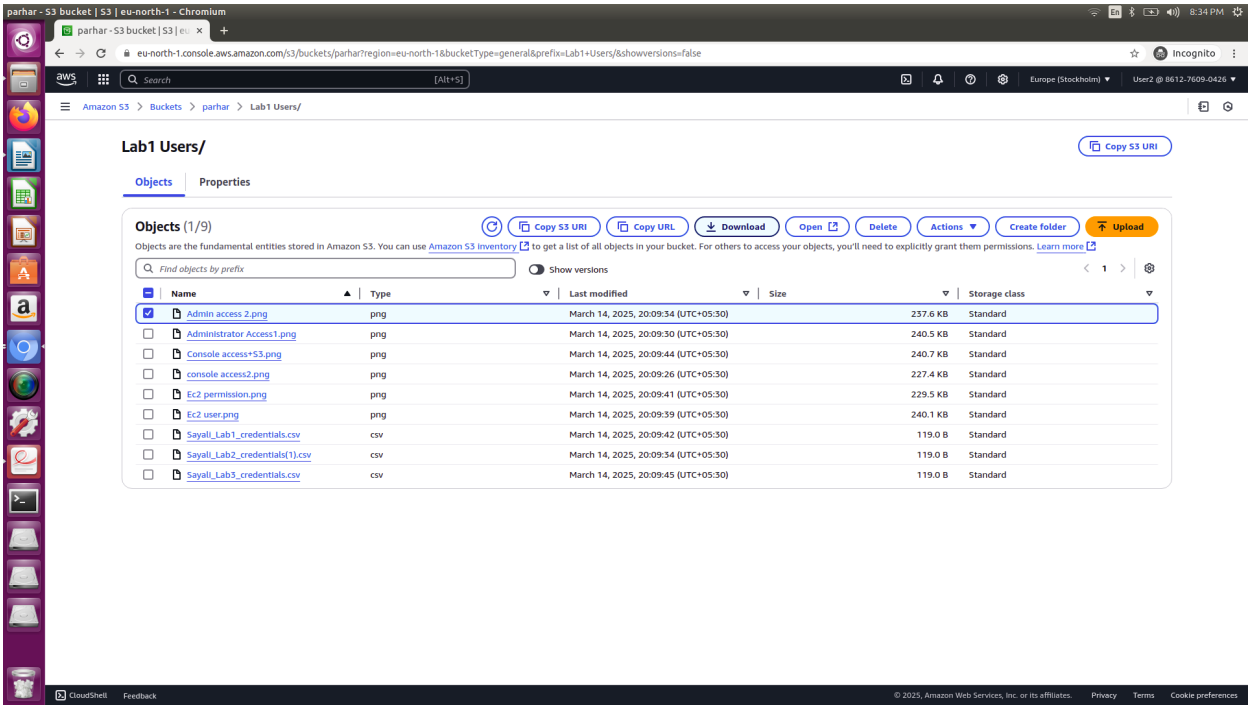

Figure 3.3: S3 write access through downloading option

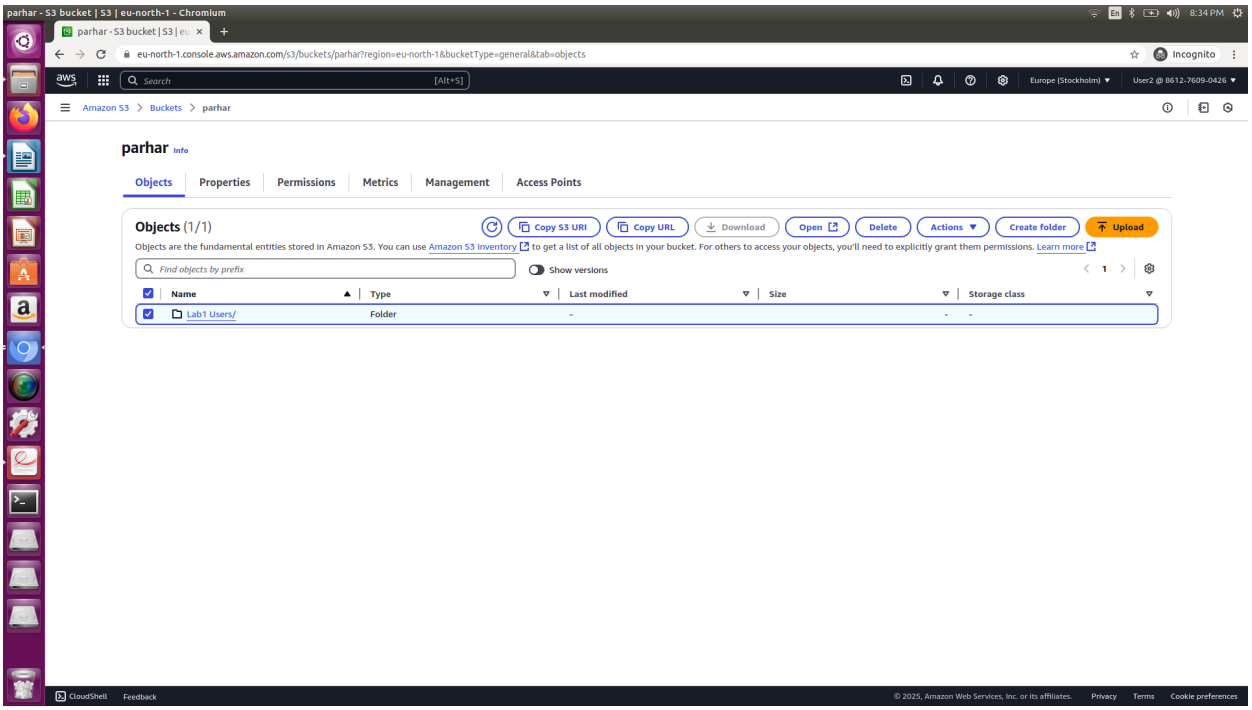

Figure 3.4: S3 write access through deleting option

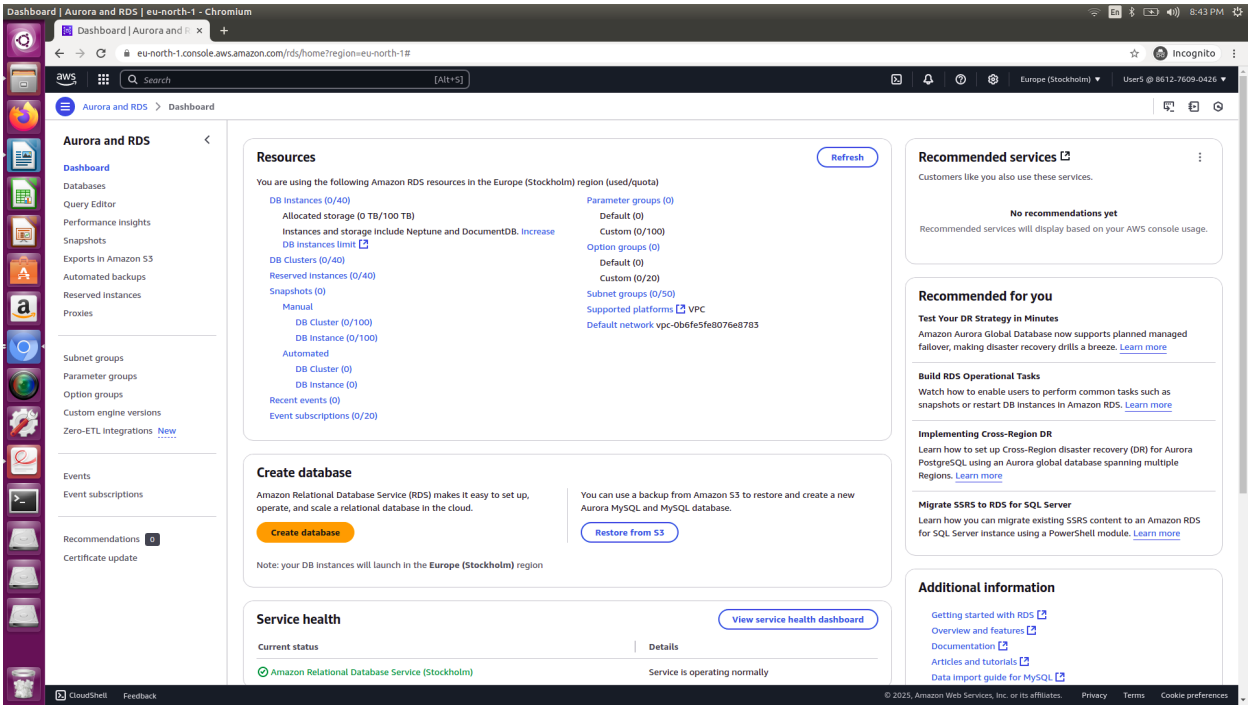

Figure 3.5: RDS Full Access Dashboard

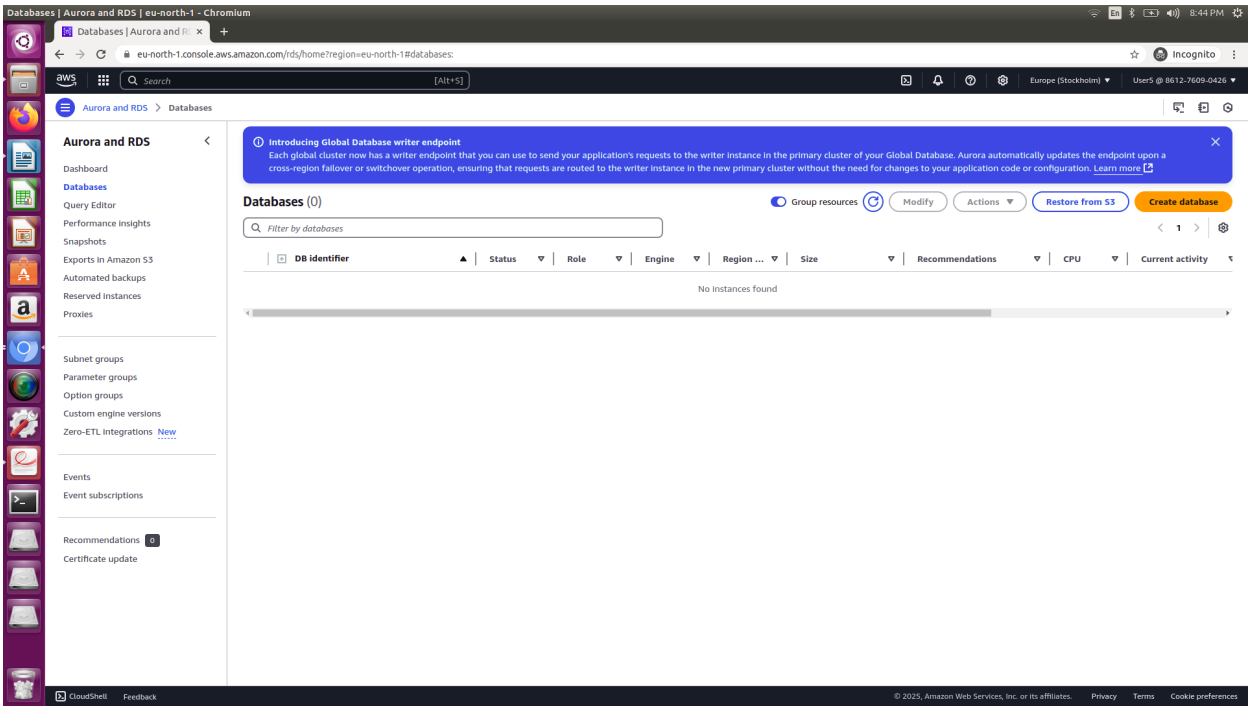

Figure 3.6: RDS Full Access DB

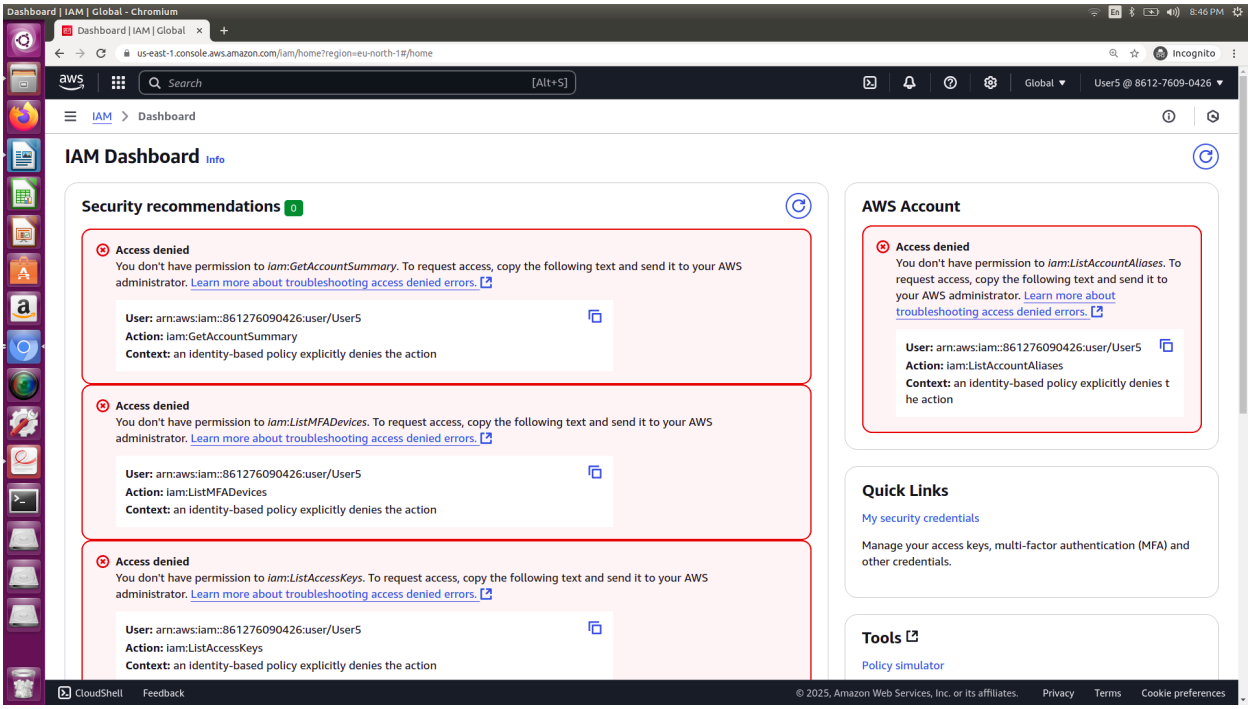

Figure 3.7: showing IAM restrict access

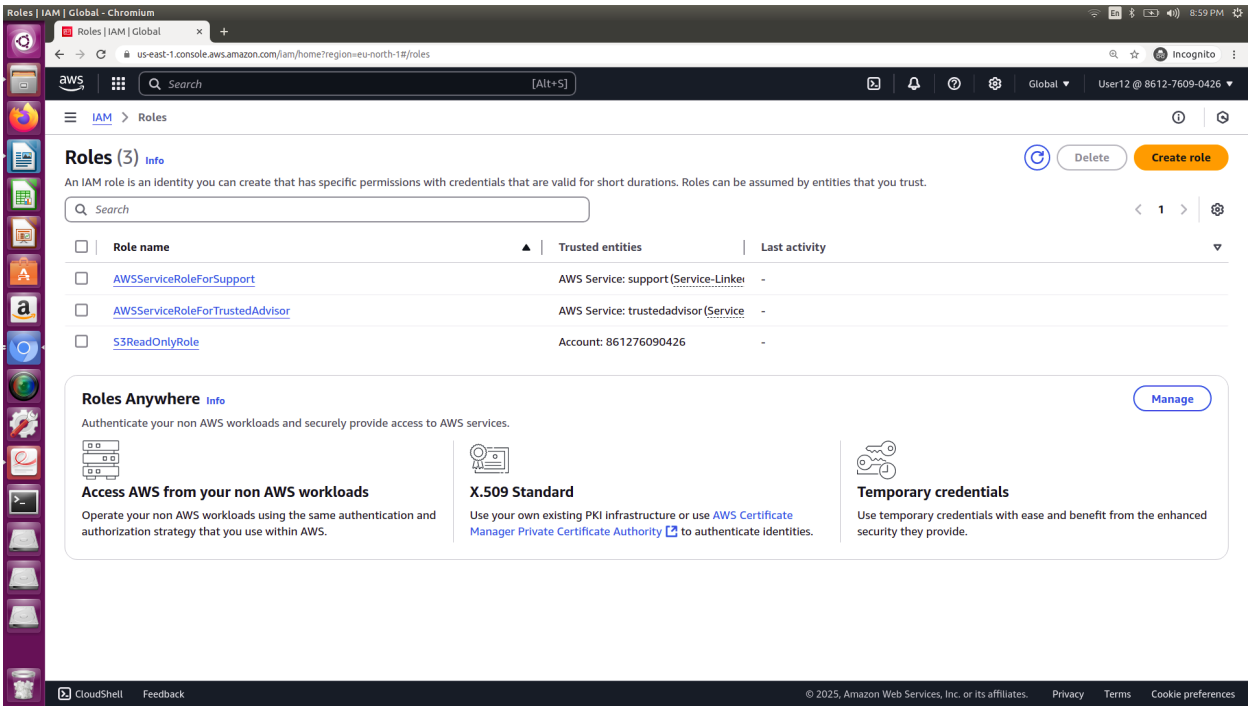

Figure 3.8: IAM Role read only access

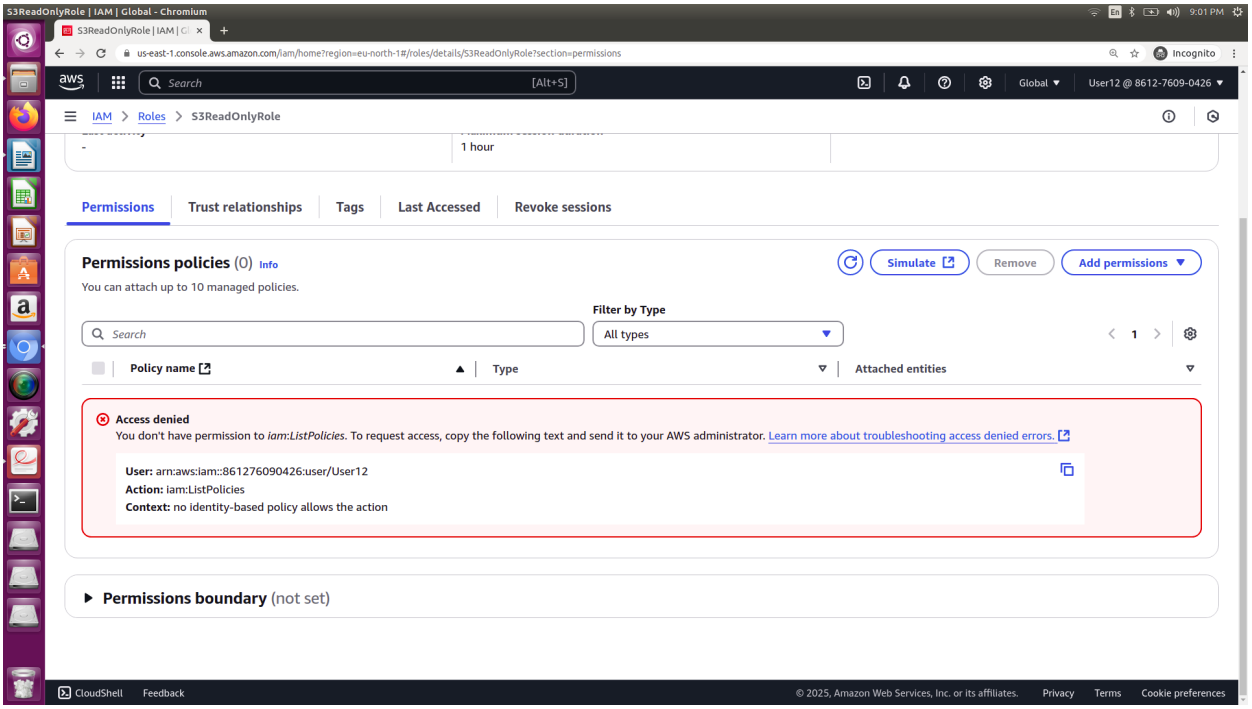

Figure 3.9: IAM Role restrict modification access

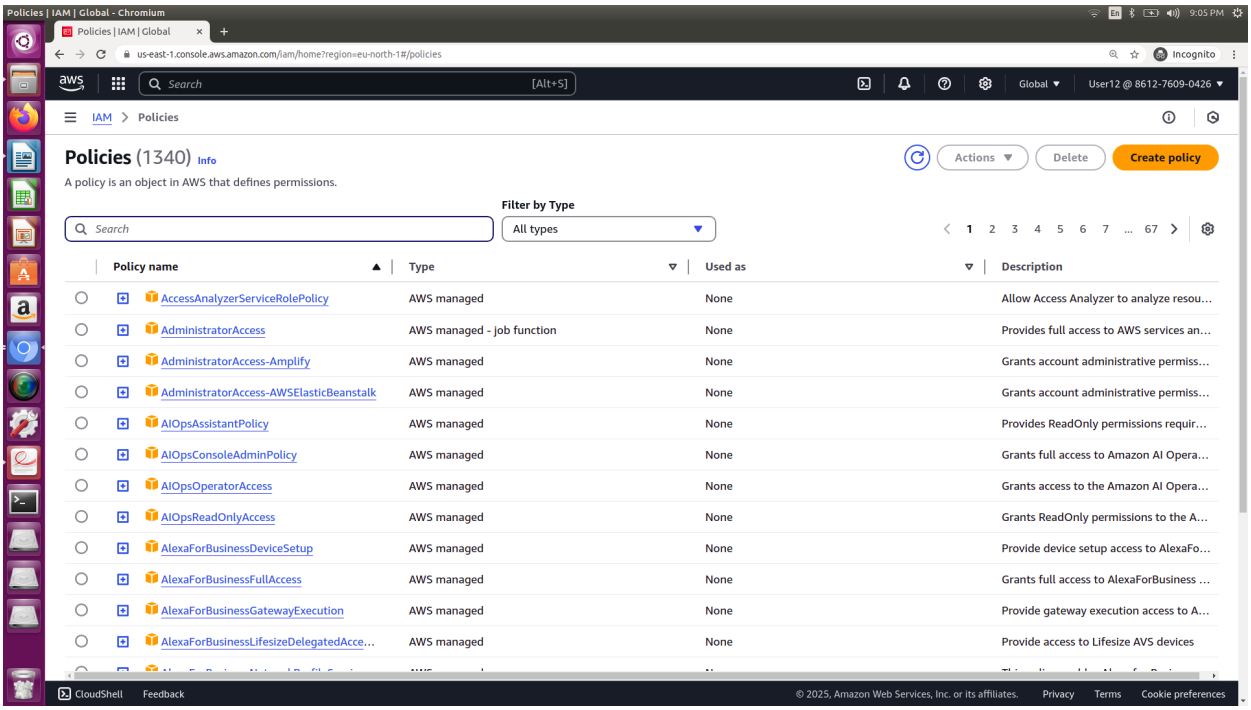

Figure 3.10: IAM Policy read only access

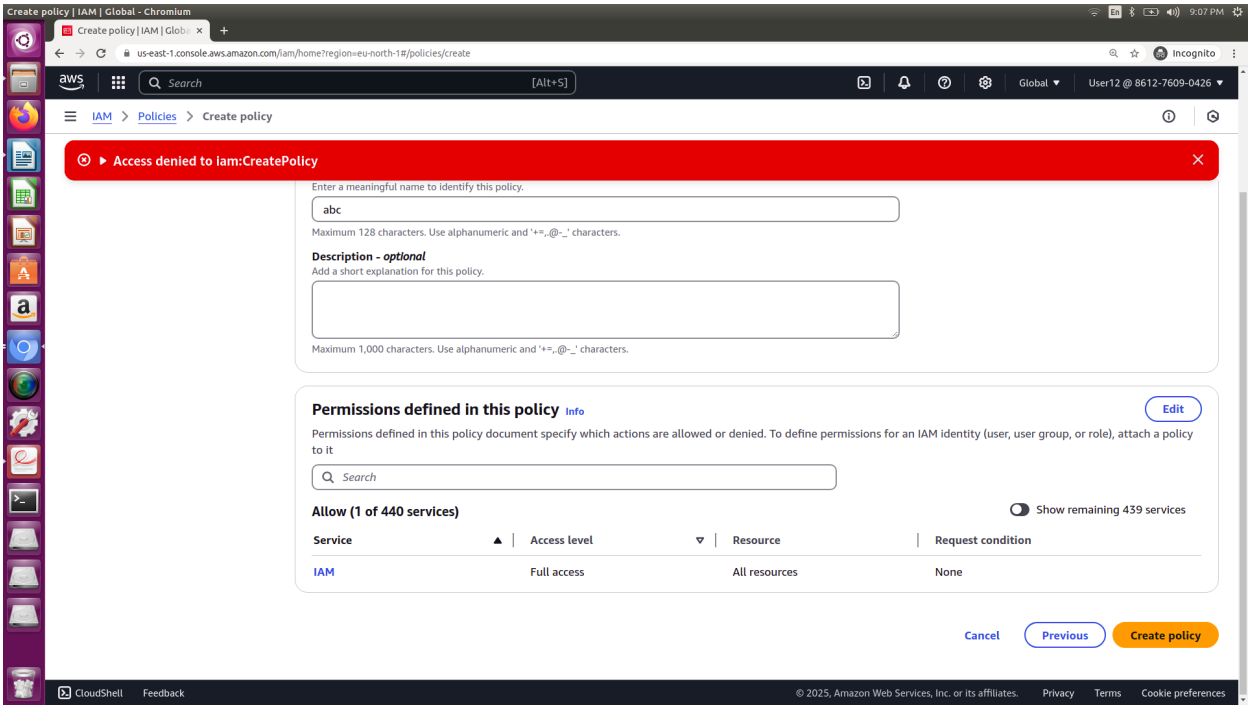

Figure 3.11: IAM Policy restrict modification access

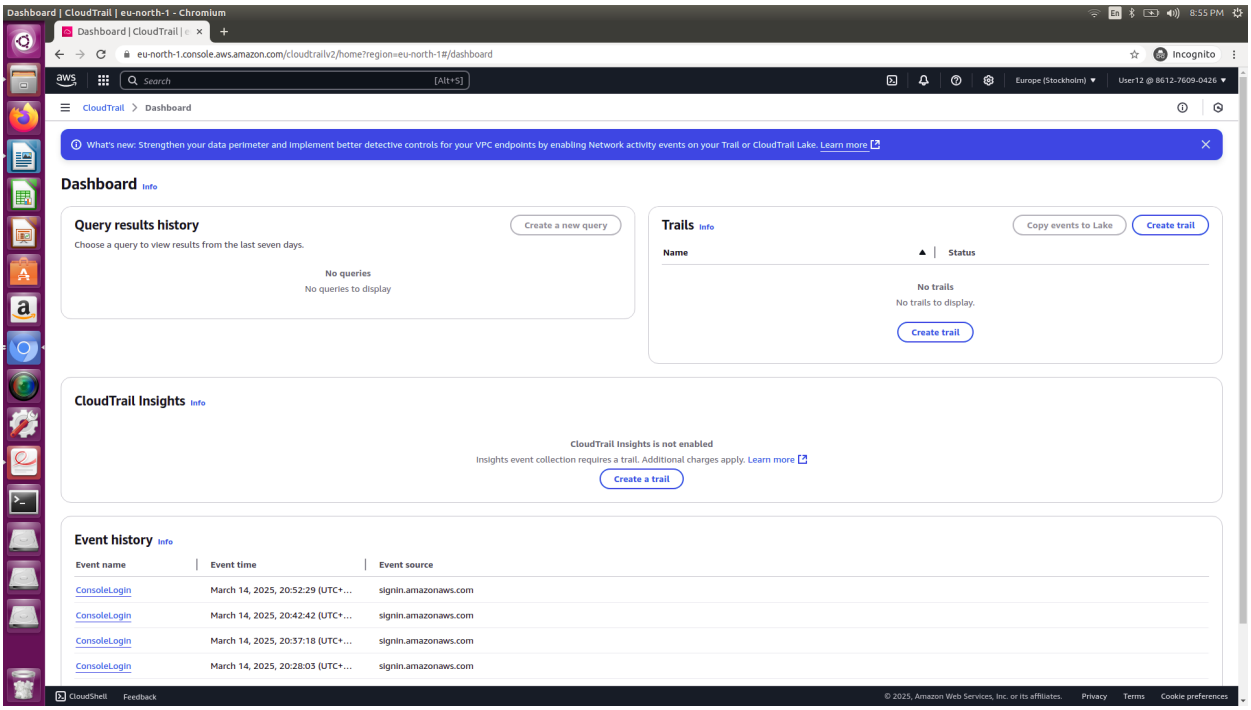

Figure 3.12: CloudTrail read only access

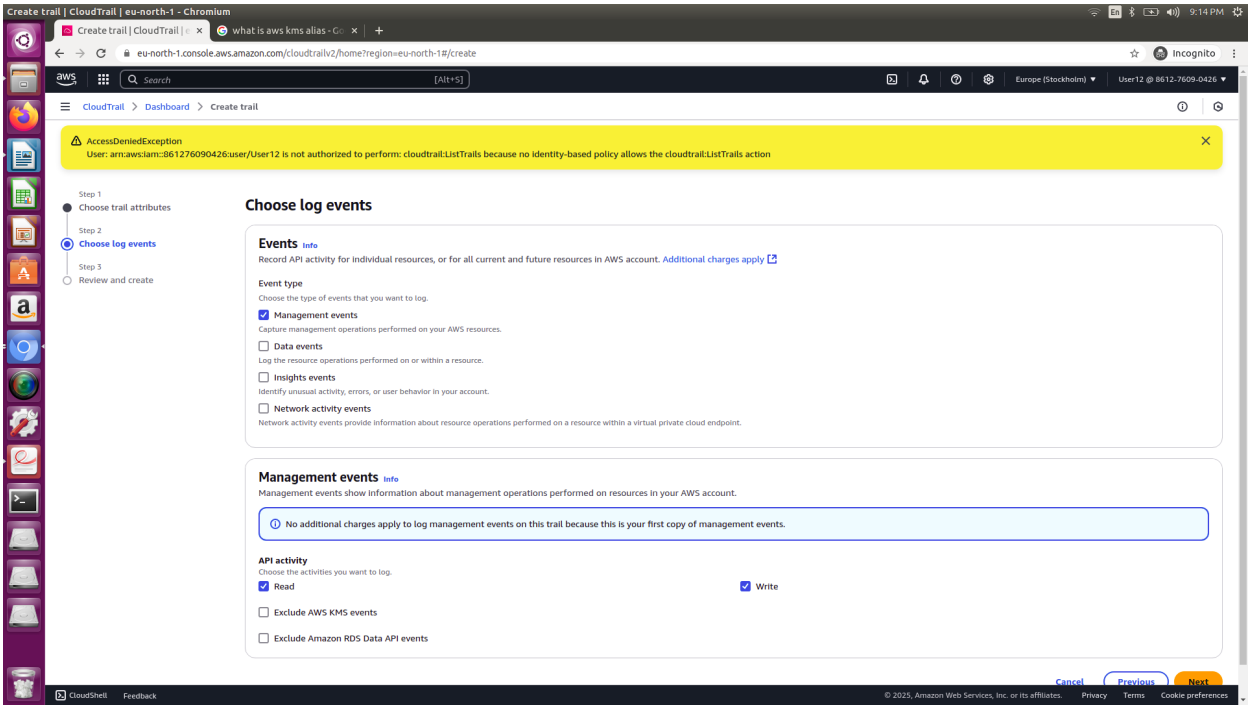

Figure 3.13: CloudTrail restrict modification access

Chapter 4

ADVANCED CHALLENGE

In Advanced challenge,I created an IAM Role (S3ReadOnlyRole) with read-only access to S3 and assume this role to Developers group using sts:AssumeRole. If the Organization want a particular user use a policy for short term then in that case, through assume role they can give access to user to use the policy for short period of time and after the time expires the user can't access the policy.

1.0 Test cases for role assumption via AWS CLI and Console

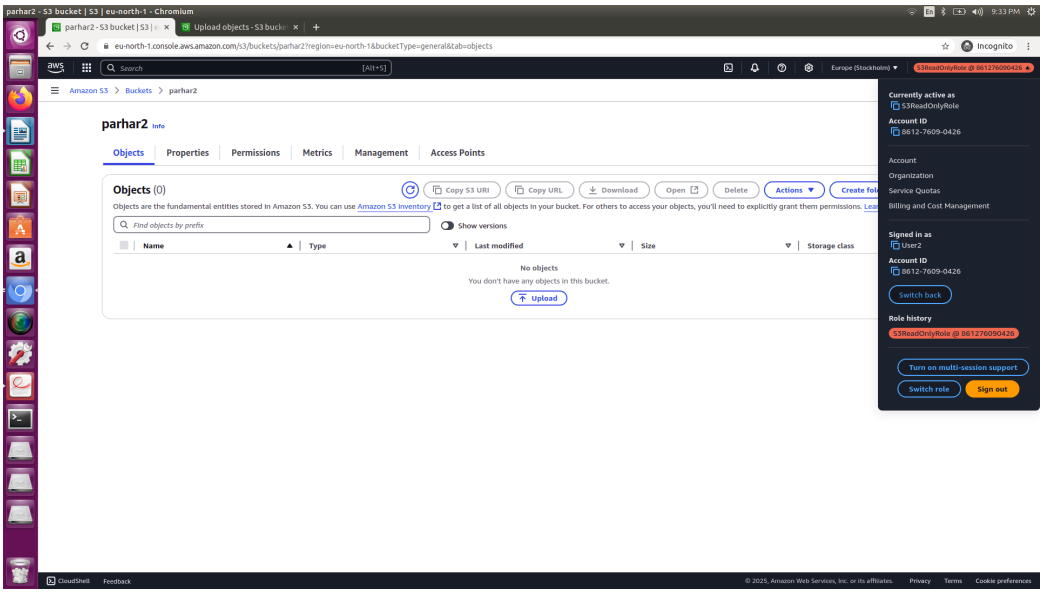

Figure 4.1: succesfully switch role to developers group and giving read only access to s3 bucket

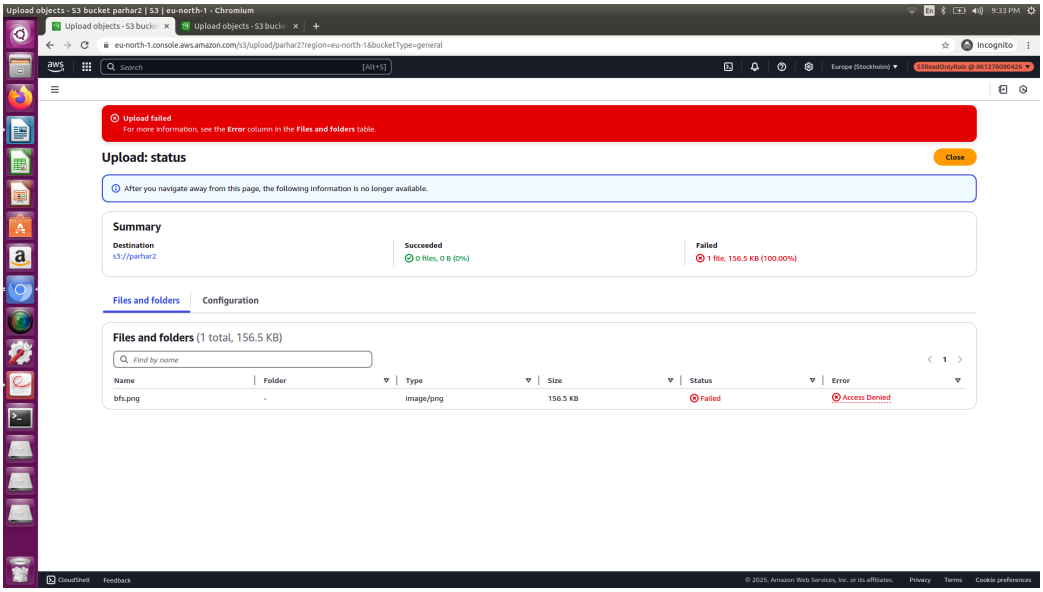

Figure 4.2: restrict modification access to s3 bucket

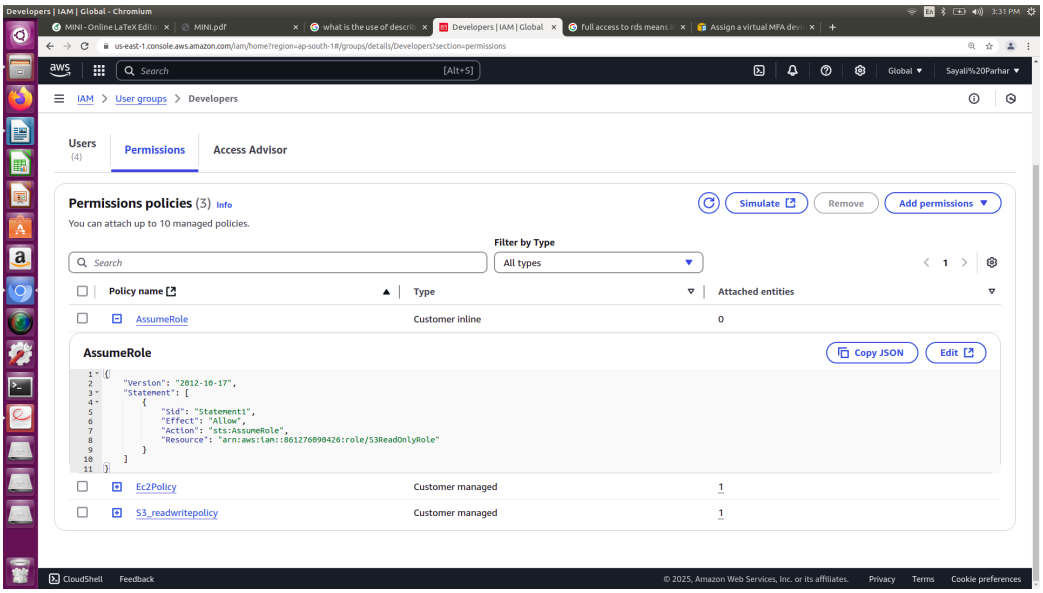

Figure 4.3: Assume role using CLI

Chapter 5

CONCLUSION

In this project, I successfully built a Secure IAM Security Model for an Organization to ensure the security of multiple teams, each with different level of access requirements such as EC2,S3,RDS and IAM Services.