Placement Empowerment Program

Cloud Computing and DevOps Centre

Set Up IAM Roles and Permissions: Create an IAM role on your cloud platform. Assign the role to your VM to restrict/allow specific actions.

Name: Jeffersen Godfrey A M **Department:** CSE

INTRODUCTION:

This Proof of Concept (PoC) demonstrates the process of setting up and utilizing IAM roles and permissions in AWS. The goal is to show how to secure AWS resources by managing access through roles rather than hardcoding credentials. Specifically, this PoC focuses on creating an IAM role, assigning it to an EC2 instance, and verifying the instance's access to AWS services such as Amazon S3.

OVERVIEW:

The process is divided into several key steps:

1. **Create an IAM Role**: Define a role in AWS IAM and attach policies that grant permissions for specific AWS services.

2. **Launch an EC2 Instance**: Create a virtual machine (VM) in AWS and configure it for testing the assigned IAM role.

3. **Assign the IAM Role to the EC2 Instance**: Attach the created IAM role to the EC2 instance to enable access to AWS services without using access keys.

4. **Verify Access**: Test the EC2 instance to confirm that it has the appropriate permissions by interacting with services like Amazon S3.

OBJECTIVES:

This PoC aims to achieve the following objectives:

1. **Secure Access**: Implement IAM roles to grant temporary permissions to AWS resources without embedding credentials.

2. **Demonstrate Role-Based Permissions**: Show how roles can restrict or allow actions based on attached policies.

3. **Test Least Privilege Principle**: Ensure that the EC2 instance only has the permissions it needs to perform specific tasks.

4. **Hands-On Learning**: Provide practical experience with IAM roles and their applications in a cloud environment.

IMPORTANCE:

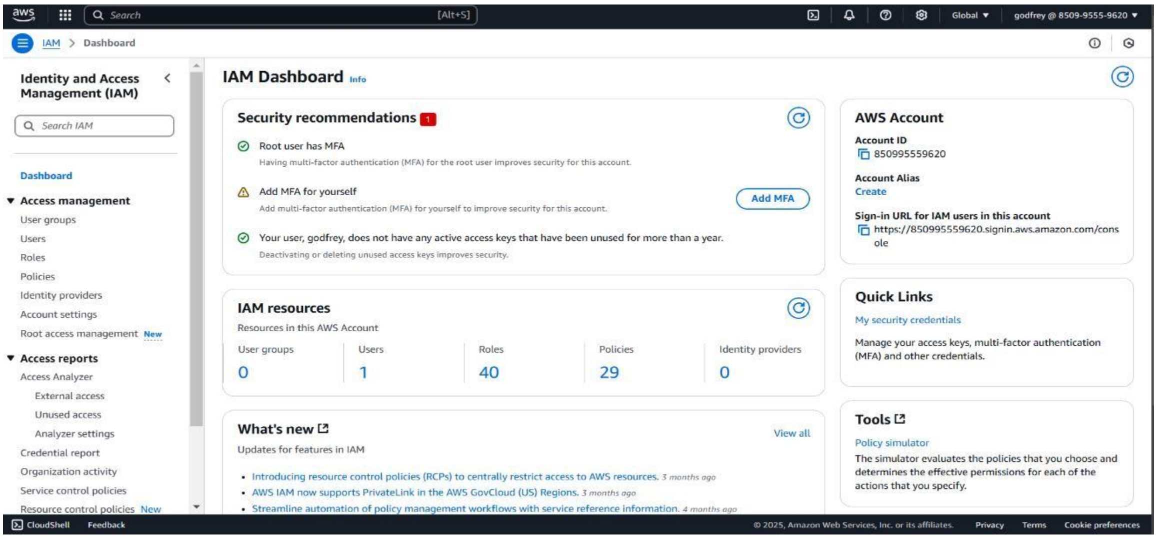
IAM roles and permissions are fundamental to securing cloud environments. They allow for fine-grained access control and improve operational efficiency by:

1. **Eliminating Hardcoded Credentials**: Reducing security risks by avoiding the storage of access keys in applications or instances.
2. **Granting Least Privilege Access**: Ensuring users and resources only have the permissions they require, minimizing potential misuse.
3. **Improving Compliance**: Enforcing organizational policies and audit requirements.
4. **Enhancing Automation**: Allowing resources like EC2 instances to securely interact with other AWS services.

STEP-BY-STEP OVERVIEW:

Step 1:

1. In the AWS Management Console, type **"IAM"** in the search bar at the top.
2. Click on **IAM** from the search results.



IAM Dashboard Info

Security recommendations 1

Root user has MFA
Having multi-factor authentication (MFA) for the root user improves security for this account.

Add MFA for yourself
Add multi-factor authentication (MFA) for yourself to improve security for this account.

Your user, godfrey, does not have any active access keys that have been unused for more than a year.
Deactivating or deleting unused access keys improves security.

IAM resources
Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
0	1	40	29	0

What's new
Updates for features in IAM

- Introducing resource control policies (RCPs) to centrally restrict access to AWS resources. 3 months ago
- AWS IAM now supports PrivateLink in the AWS GovCloud (US) Regions. 3 months ago
- Streamline automation of policy management workflows with service reference information. 4 months ago

AWS Account

Account ID
850995559620

Account Alias
Create

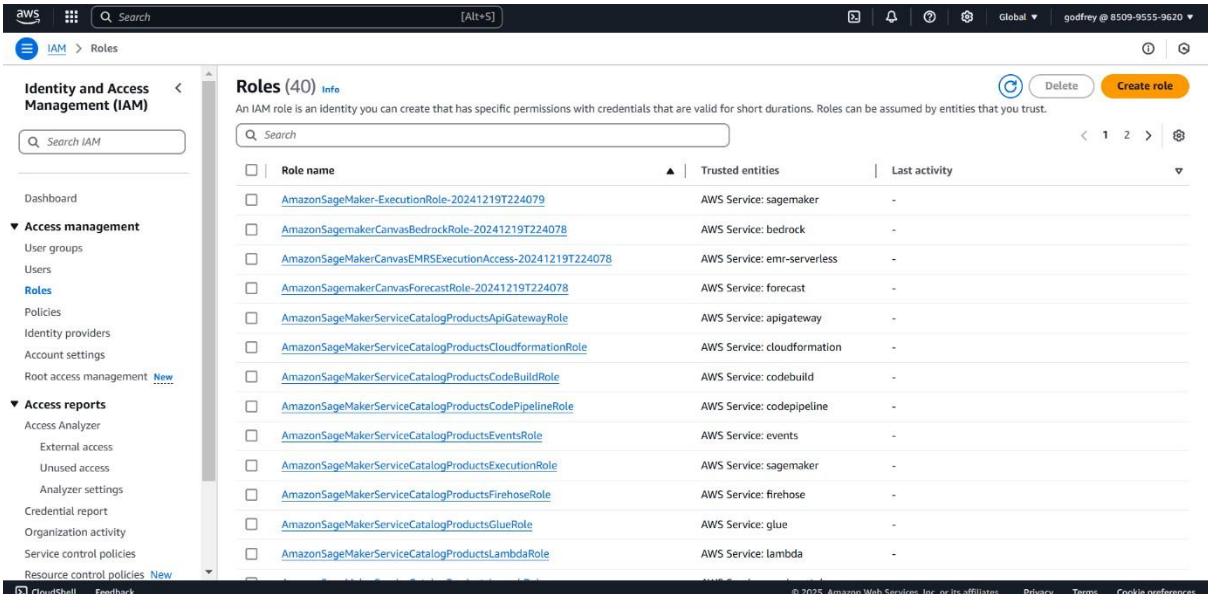
Sign-in URL for IAM users in this account
https://850995559620.signin.aws.amazon.com/console

Quick Links
My security credentials
Manage your access keys, multi-factor authentication (MFA) and other credentials.

Tools
Policy simulator
The simulator evaluates the policies that you choose and determines the effective permissions for each of the actions that you specify.

Step 2:

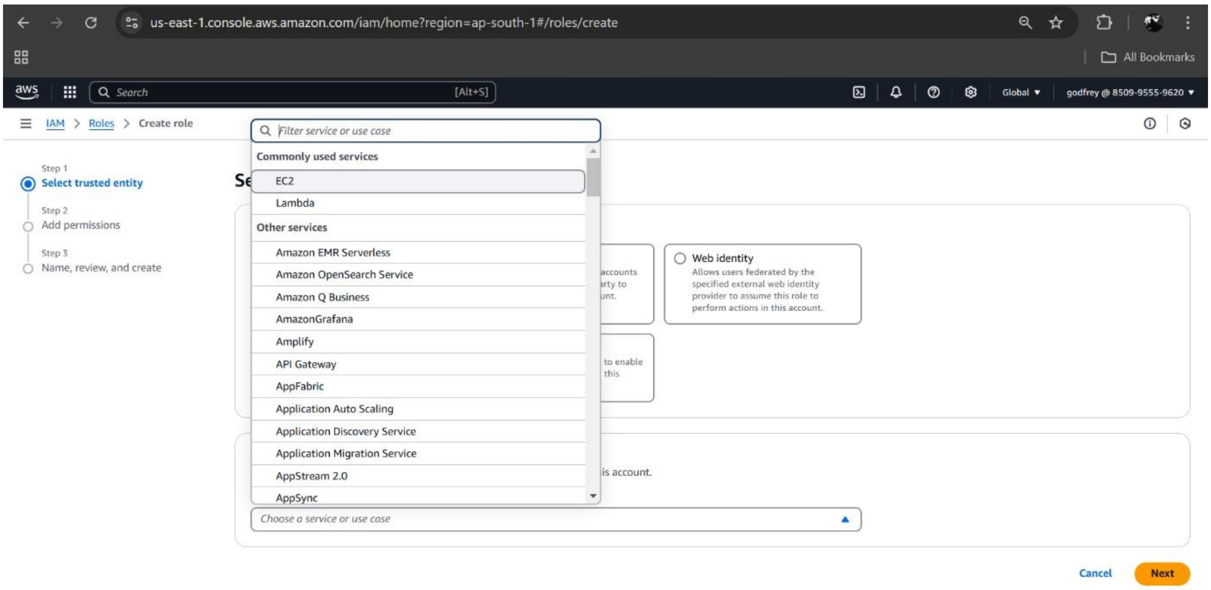
1. On the IAM dashboard, click on **"Roles"** in the lefthand menu.
2. On the Roles page, click the **"Create Role"** button.



Roles (40) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

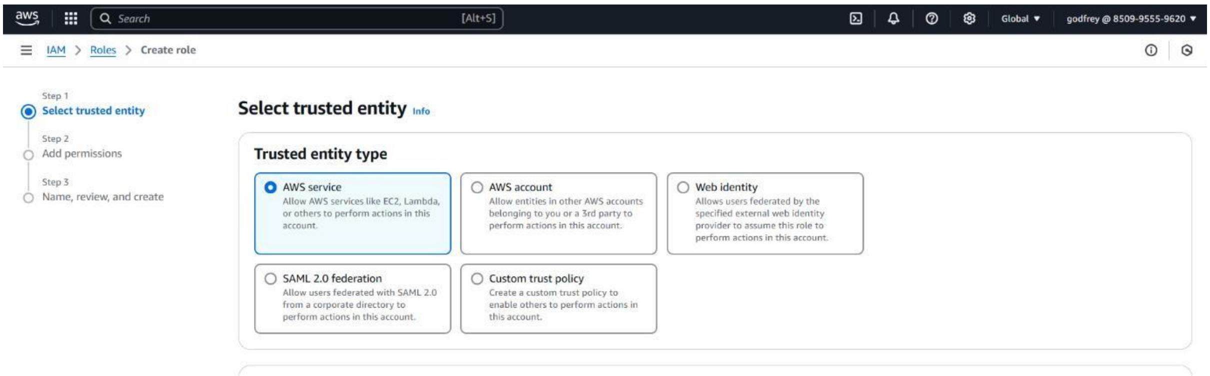
□	Role name	Trusted entities	Last activity
□	AmazonSageMaker-ExecutionRole-20241219T224079	AWS Service: sagemaker	-
□	AmazonSagemakerCanvasBedrockRole-20241219T224078	AWS Service: bedrock	-
□	AmazonSageMakerCanvasEMRSExecutionAccess-20241219T224078	AWS Service: emr-serverless	-
□	AmazonSagemakerCanvasForecastRole-20241219T224078	AWS Service: forecast	-
□	AmazonSageMakerServiceCatalogProductsApiGatewayRole	AWS Service: apigateway	-
□	AmazonSageMakerServiceCatalogProductsCloudformationRole	AWS Service: cloudformation	-
□	AmazonSageMakerServiceCatalogProductsCodeBuildRole	AWS Service: codebuild	-
□	AmazonSageMakerServiceCatalogProductsCodePipelineRole	AWS Service: codepipeline	-
□	AmazonSageMakerServiceCatalogProductsEventsRole	AWS Service: events	-
□	AmazonSageMakerServiceCatalogProductsExecutionRole	AWS Service: sagemaker	-
□	AmazonSageMakerServiceCatalogProductsFirehoseRole	AWS Service: firehose	-
□	AmazonSageMakerServiceCatalogProductsGlueRole	AWS Service: glue	-
□	AmazonSageMakerServiceCatalogProductsLambdaRole	AWS Service: lambda	-


us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles/create
All Bookmarks
aws
Search
[Alt+S]
Global ▾
godfrey @ 8509-9555-9620 ▾
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Filter service or use case
Commonly used services
EC2
Lambda
Other services
Amazon EMR Serverless
Amazon OpenSearch Service
Amazon Q Business
AmazonGrafana
Amplify
API Gateway
AppFabric
Application Auto Scaling
Application Discovery Service
Application Migration Service
AppStream 2.0
AppSync
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
Choose a service or use case
Cancel
Next

Step 3:

1. On the **"Create Role"** page, under **Trusted Entity Type**, select **AWS Service** (it should be selected by default).
2. In the **Use Case** dropdown, choose **EC2**.

Click **Next** to continue


aws
Search
[Alt+S]
Global ▾
godfrey @ 8509-9555-9620 ▾
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity Info
Trusted entity type
● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Step 4:

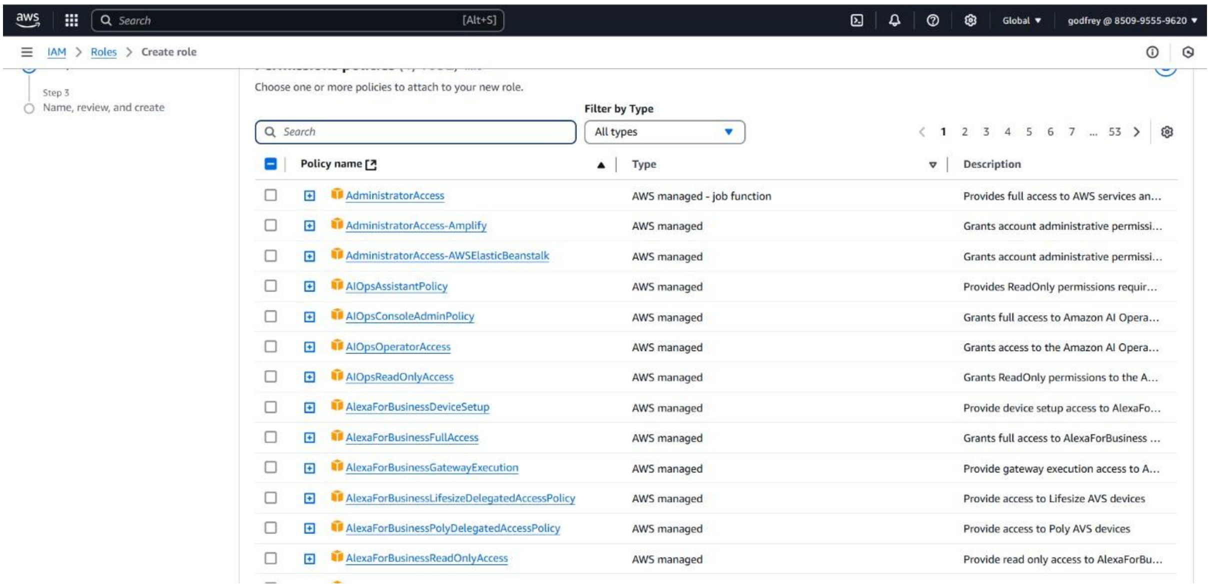
1. On the **Permissions** page, you’ll see a list of policies.

2. Select a policy based on what actions you want the VM to perform. For example:

   To give the VM **read-only access to S3**, select **AmazonS3ReadOnlyAccess**.

   You can search for policies in the search bar (e.g., type "S3" for S3 policies).

3. Once you've selected a policy, click **Next**.



Choose one or more policies to attach to your new role.

Filter by Type: All types

Search

< 1 2 3 4 5 6 7 ... 53 >

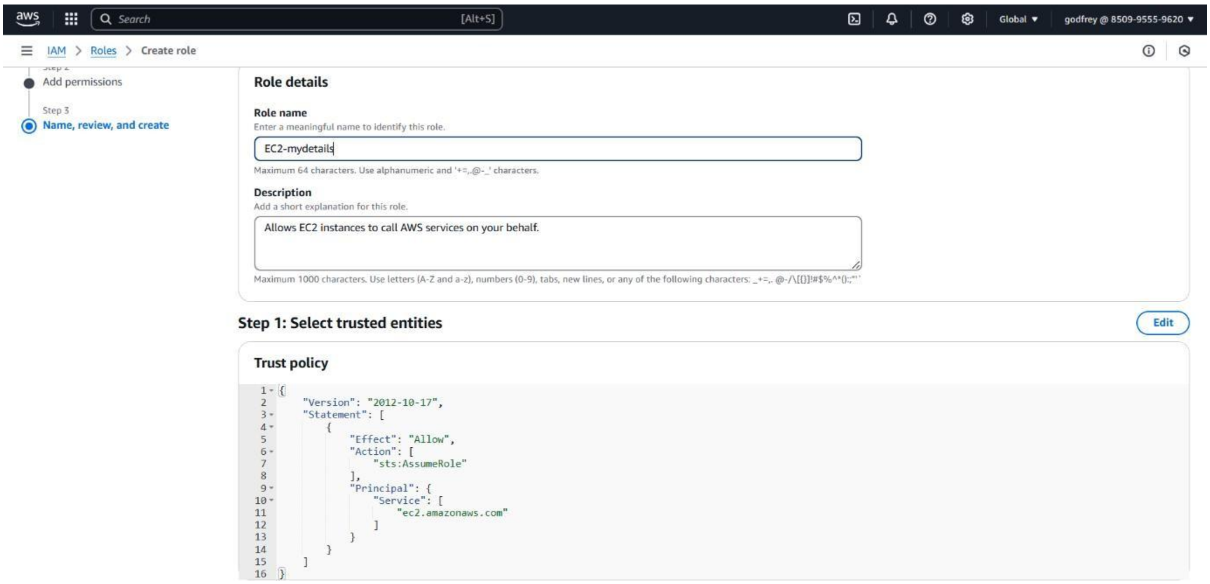
	Policy name	Type	Description
□	AdministratorAccess	AWS managed - job function	Provides full access to AWS services an...
□	AdministratorAccess-Amplify	AWS managed	Grants account administrative permissi...
□	AdministratorAccess-AWSElasticBeanstalk	AWS managed	Grants account administrative permissi...
□	AIOpsAssistantPolicy	AWS managed	Provides ReadOnly permissions requir...
□	AIOpsConsoleAdminPolicy	AWS managed	Grants full access to Amazon AI Opera...
□	AIOpsOperatorAccess	AWS managed	Grants access to the Amazon AI Opera...
□	AIOpsReadOnlyAccess	AWS managed	Grants ReadOnly permissions to the A...
□	AlexaForBusinessDeviceSetup	AWS managed	Provide device setup access to AlexaFo...
□	AlexaForBusinessFullAccess	AWS managed	Grants full access to AlexaForBusiness ...
□	AlexaForBusinessGatewayExecution	AWS managed	Provide gateway execution access to A...
□	AlexaForBusinessLifesizeDelegatedAccessPolicy	AWS managed	Provide access to Lifesize AVS devices
□	AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	Provide access to Poly AVS devices
□	AlexaForBusinessReadOnlyAccess	AWS managed	Provide read only access to AlexaForBu...

Step 5:

1. On the **Role Details** page:

- Enter a name for your role (e.g., My-EC2-S3Access-Role).
- (Optional) Add a description or tags if you’d like.

2. Click **Create Role** to finish.



Role details

Role name

Enter a meaningful name to identify this role.

EC2-mydetails

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Description

Add a short explanation for this role.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/\[{}]!#$%^*();:"'`

Step 1: Select trusted entities

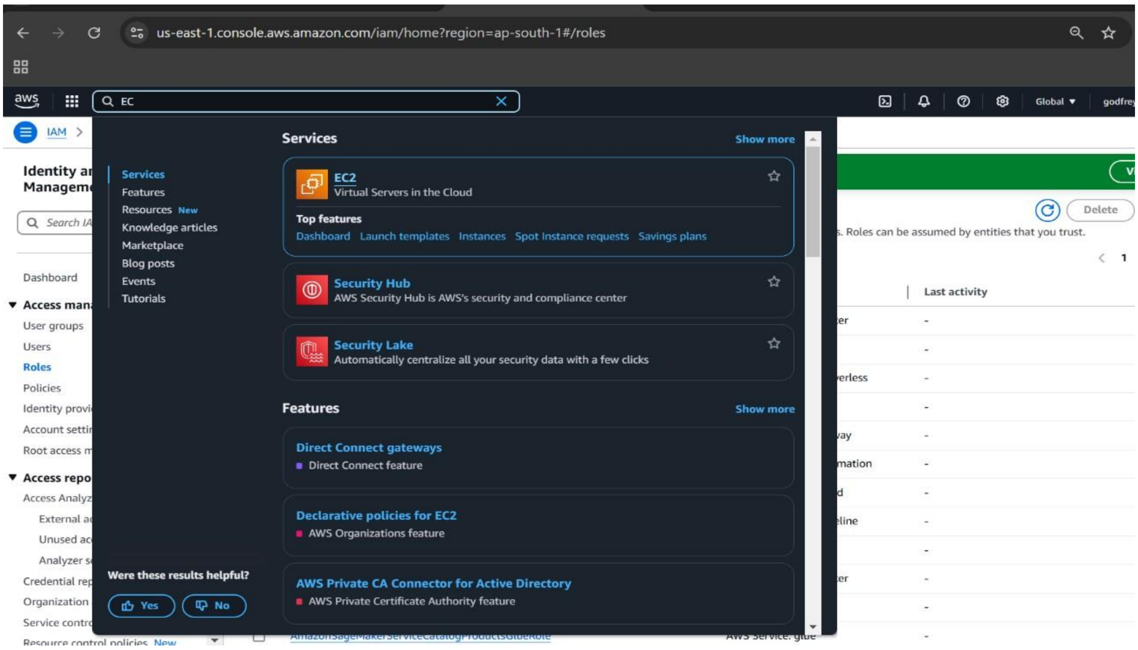
Trust policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sts:AssumeRole"
            ],
            "Principal": {
                "Service": [
                    "ec2.amazonaws.com"
                ]
            }
        }
    ]
}
```

Step 6:

1. In the AWS Management Console, search for **EC2** and click to open the **EC2 Dashboard**.

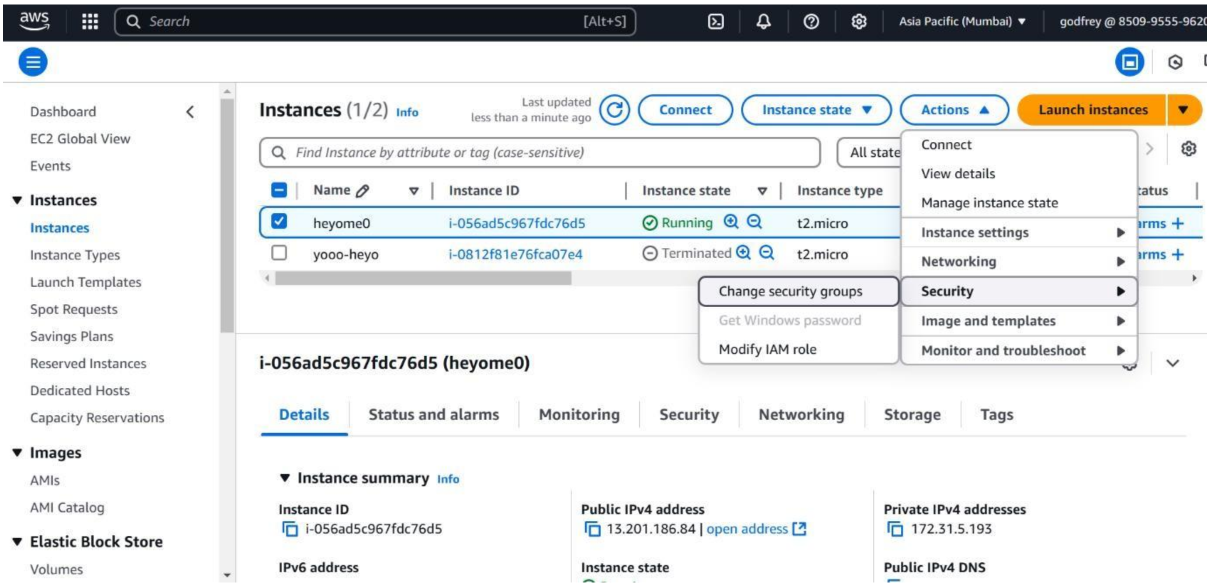
2. Select the instance (VM) you want to assign the IAM role to.



Step 7:

1. In the **Instance details** section, click **Actions** in the top right corner.

2. From the dropdown, choose **Security** > **Modify IAM Role**.



☐	Name	Instance ID	Instance state	Instance type
☑	heyome0	i-056ad5c967fdc76d5	Running	t2.micro
☐	yooo-heyo	i-0812f81e76fca07e4	Terminated	t2.micro

i-056ad5c967fdc76d5 (heyome0)

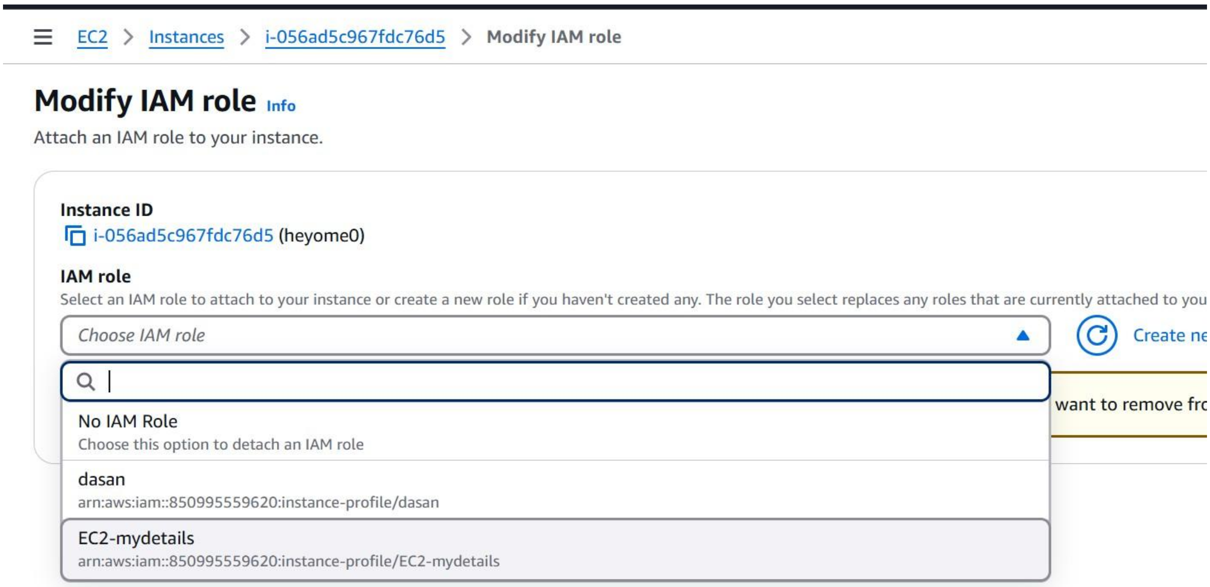
Instance ID: i-056ad5c967fdc76d5

Public IPv4 address: 13.201.186.84 | open address

Private IPv4 addresses: 172.31.5.193

Step 8:

1. In the **Modify IAM role** window, you should see a dropdown for **IAM role**.
2. Select the role you created earlier (e.g., My-EC2-S3Access-Role).
3. Click **Update IAM role** to apply the changes.



EC2 > Instances > i-056ad5c967fdc76d5 > Modify IAM role

Modify IAM role Info

Attach an IAM role to your instance.

Instance ID

i-056ad5c967fdc76d5 (heyome0)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to you

Choose IAM role

Create ne

want to remove fro

No IAM Role

Choose this option to detach an IAM role

dasan

arn:aws:iam::850995559620:instance-profile/dasan

EC2-mydetails

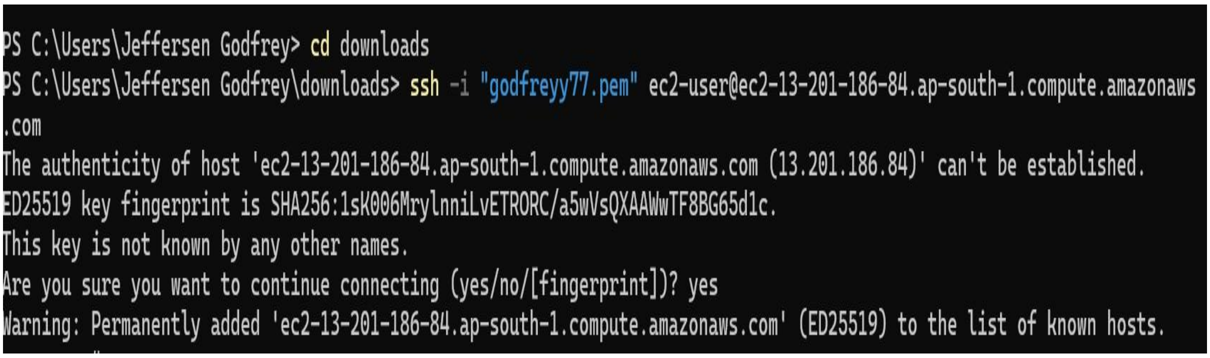
arn:aws:iam::850995559620:instance-profile/EC2-mydetails

Step 9:

1. Open your terminal (if you're using Linux or macOS) or Command Prompt (Windows).

2. Use SSH to log in to your EC2 instance. For example:

ssh -i "your-key-pair.pem" ec2-user@your-ec2-publicip

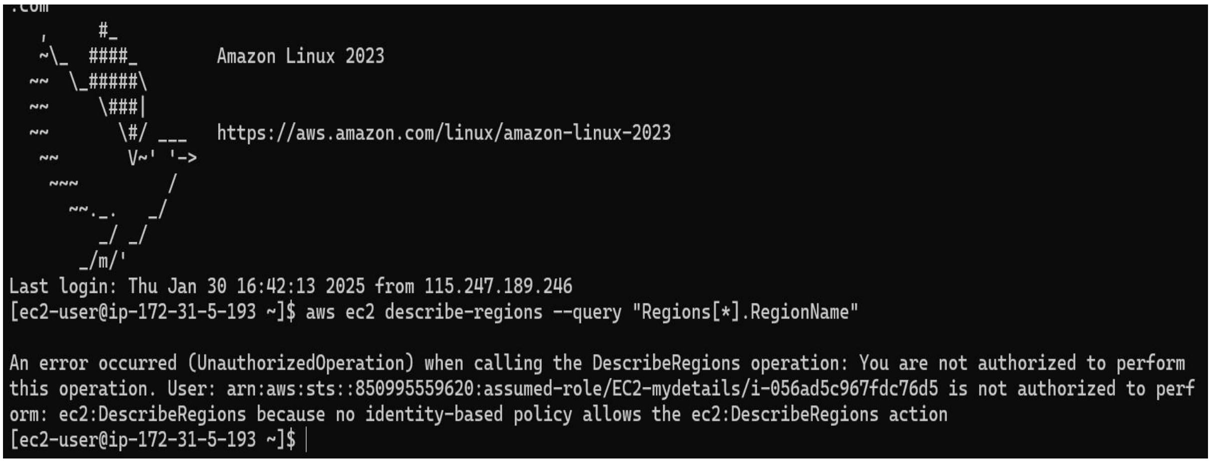


```
PS C:\Users\Jeffersen Godfrey> cd downloads
PS C:\Users\Jeffersen Godfrey\downloads> ssh -i "godfreyy77.pem" ec2-user@ec2-13-201-186-84.ap-south-1.compute.amazonaws
.com
The authenticity of host 'ec2-13-201-186-84.ap-south-1.compute.amazonaws.com (13.201.186.84)' can't be established.
ED25519 key fingerprint is SHA256:1sK006MrylnniLvETRORC/a5wVsQXAAWwTF8BG65d1c.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-201-186-84.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
```

Step 10:

[ec2-user@ip-172-31-80-54 ~]$ **aws ec2 describeregions --query "Regions[*].RegionName"**

The error confirms that your IAM role (My-EC2-S3Access-Role) does not have permissions to perform the **ec2:DescribeRegions** action. The role currently only has S3-related permissions (e.g., AmazonS3ReadOnlyAccess) and doesn't include broader EC2 permissions.



```
.com
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Thu Jan 30 16:42:13 2025 from 115.247.189.246
[ec2-user@ip-172-31-5-193 ~]$ aws ec2 describe-regions --query "Regions[*].RegionName"

An error occurred (UnauthorizedOperation) when calling the DescribeRegions operation: You are not authorized to perform
this operation. User: arn:aws:sts::850995559620:assumed-role/EC2-mydetails/i-056ad5c967fdc76d5 is not authorized to perf
orm: ec2:DescribeRegions because no identity-based policy allows the ec2:DescribeRegions action
[ec2-user@ip-172-31-5-193 ~]$
```

OUTCOME:

By completing this PoC of setting up IAM roles and permissions with an EC2 instance, you will:

1. Create an IAM role and attach policies to control access to specific AWS services.

2. Launch and configure an EC2 instance for testing purposes.

3. Assign the IAM role to the EC2 instance securely without using access keys.

4. Verify permissions by interacting with AWS services (e.g., listing S3 buckets) from the EC2 instance.

5. Demonstrate the principle of least privilege by ensuring only necessary permissions are granted.