**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: Harivasan S

Department: ADS



# Introduction:

# 

IAM (Identity and Access Management) roles in AWS enable secure and controlled access to AWS services without relying on long-term credentials. Assigning IAM roles to EC2 instances allows applications to communicate with services like S3 based on predefined permissions. This enhances security, simplifies access management, and enforces the Principle of Least Privilege. Configuring IAM roles correctly helps restrict unauthorized actions while ensuring smooth resource access within AWS environments.

# Objective:

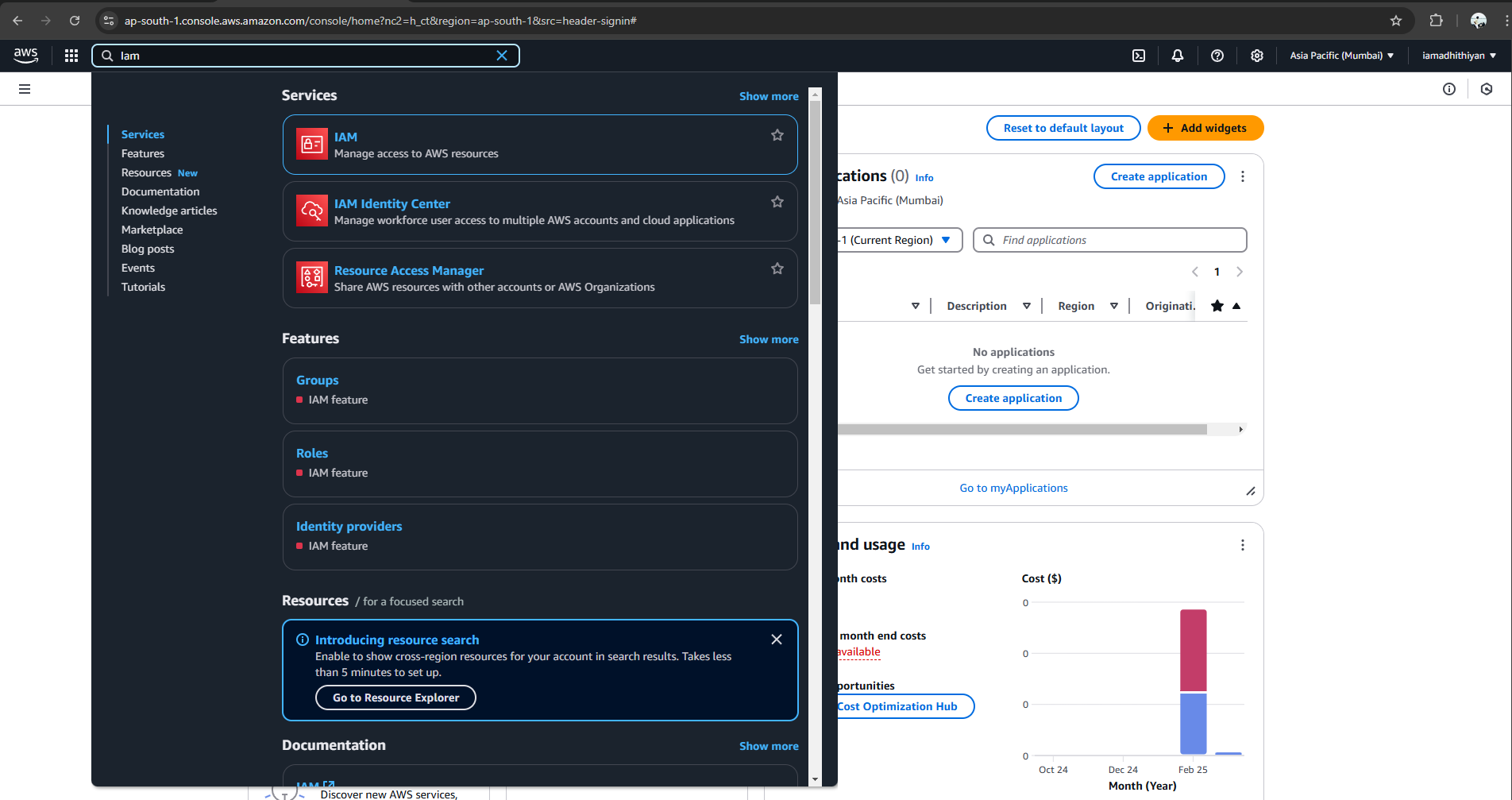
# 

1. **Enhanced Security:** Enable AWS resources like EC2 instances to communicate with other services such as S3 without relying on embedded credentials.
2. **Restricted Access Control:** Define precise permissions to regulate actions an instance or user can perform, minimizing unauthorized access or changes.
3. **Stronger Compliance & Protection:** Implement best practices like the Principle of Least Privilege to reduce security risks and meet compliance standards.

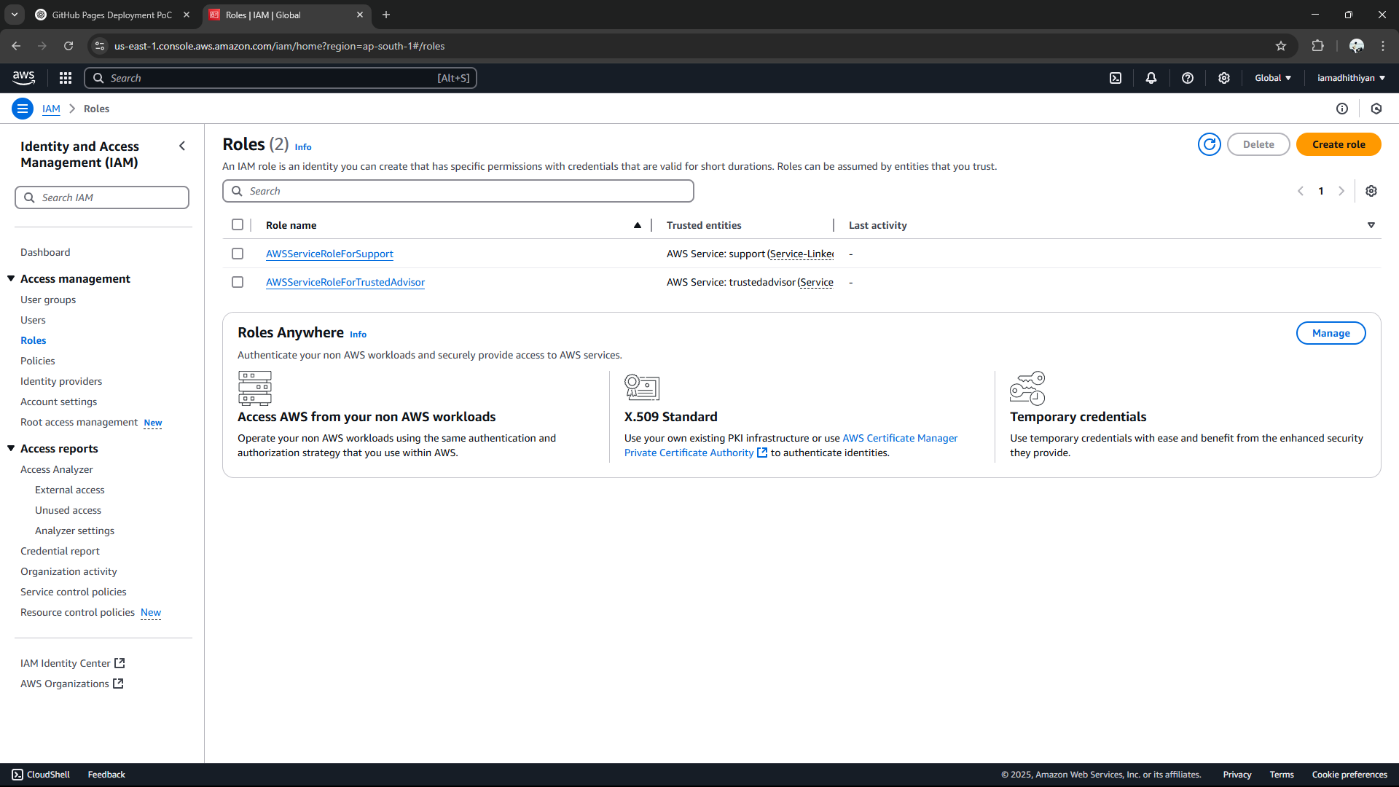
**Step-by-Step Overview**

# Step1: Create an IAM Role

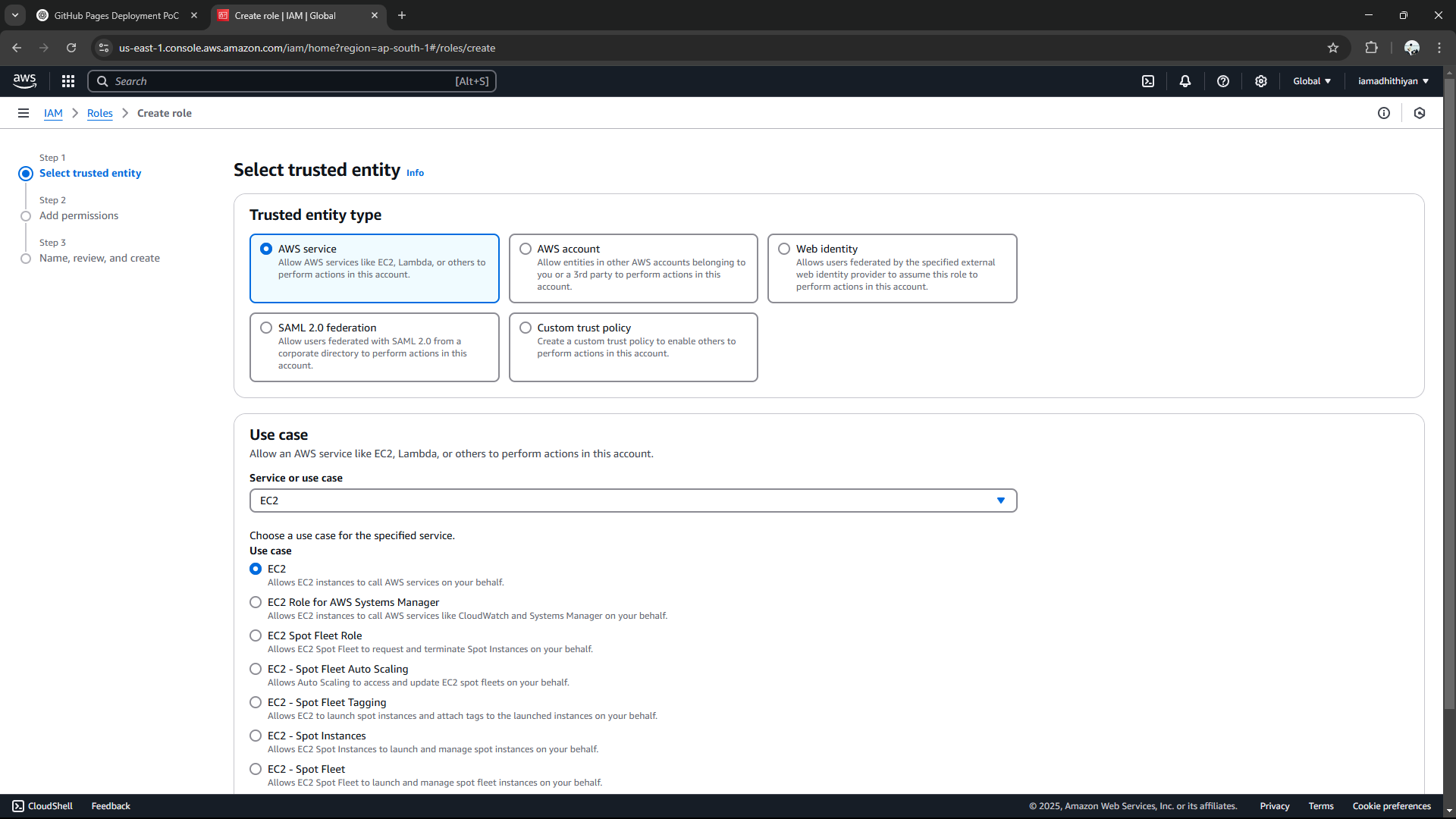
1. Sign in to your [AWS Management Console.](https://aws.amazon.com/console/) Open **IAM (Identity & Access Management)**.

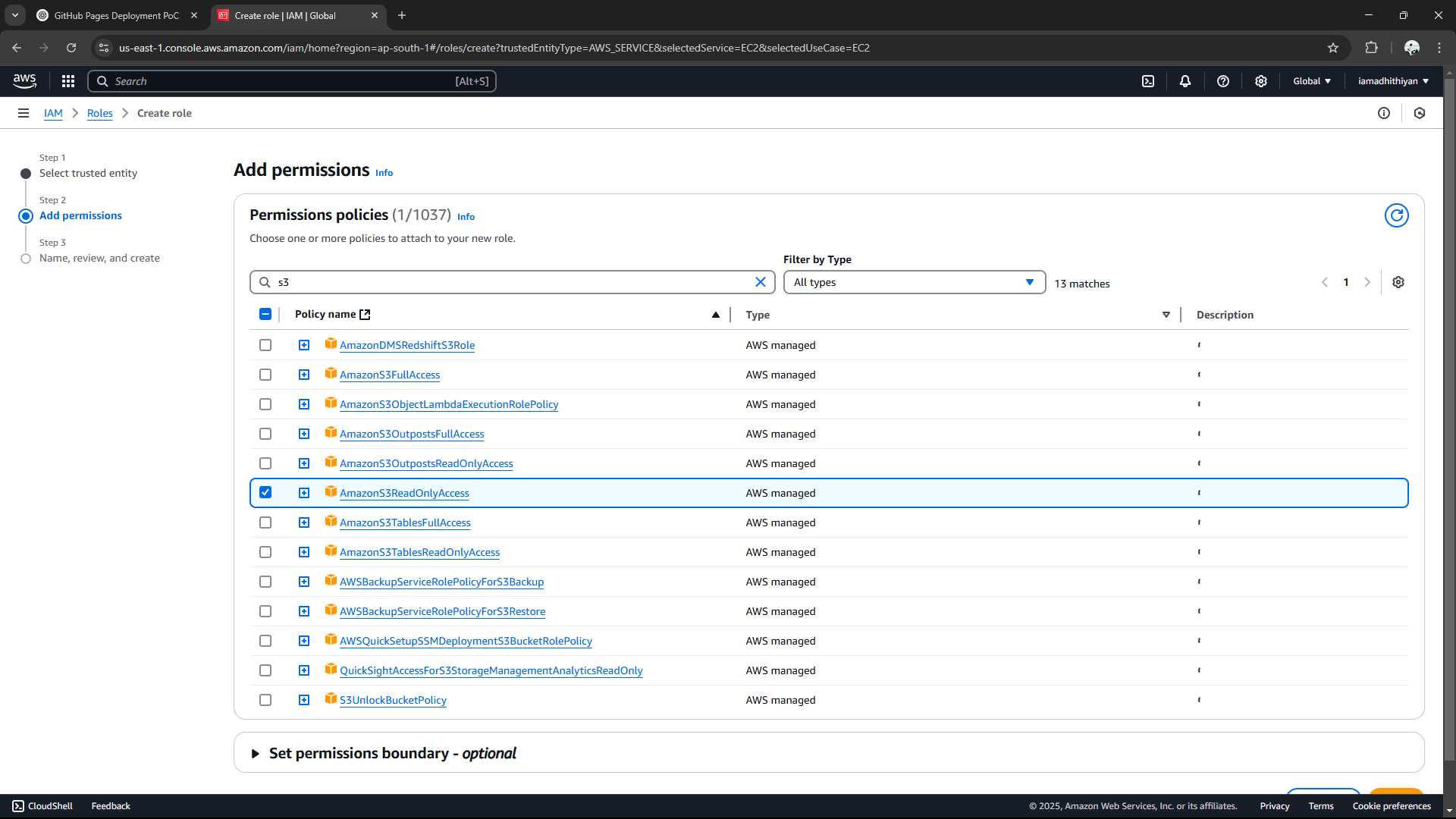


1. **Create a New Role**

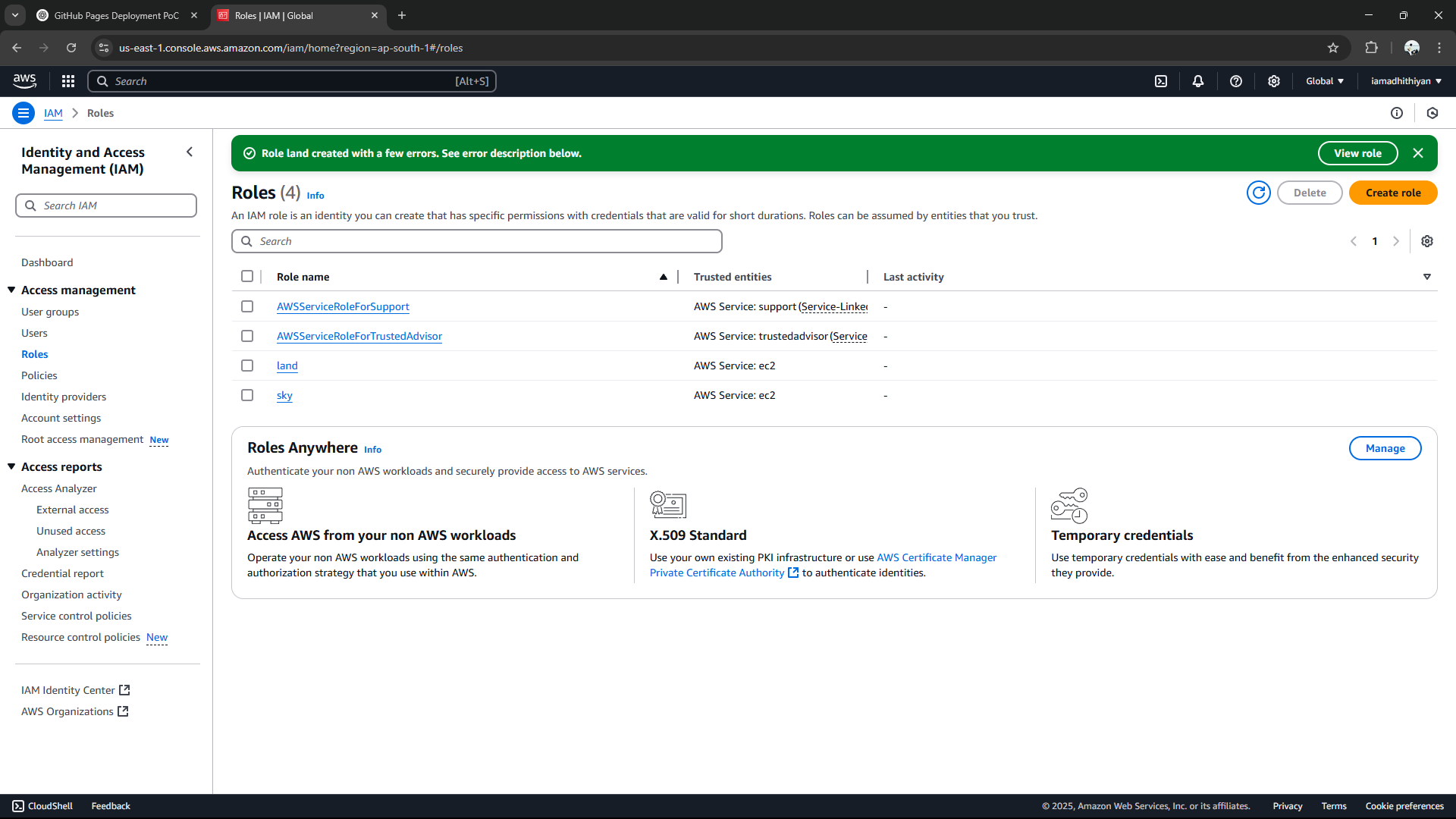


1. **Attach Permissions** : Search for **AmazonS3FullAccess** Select the policy and click **Next**.





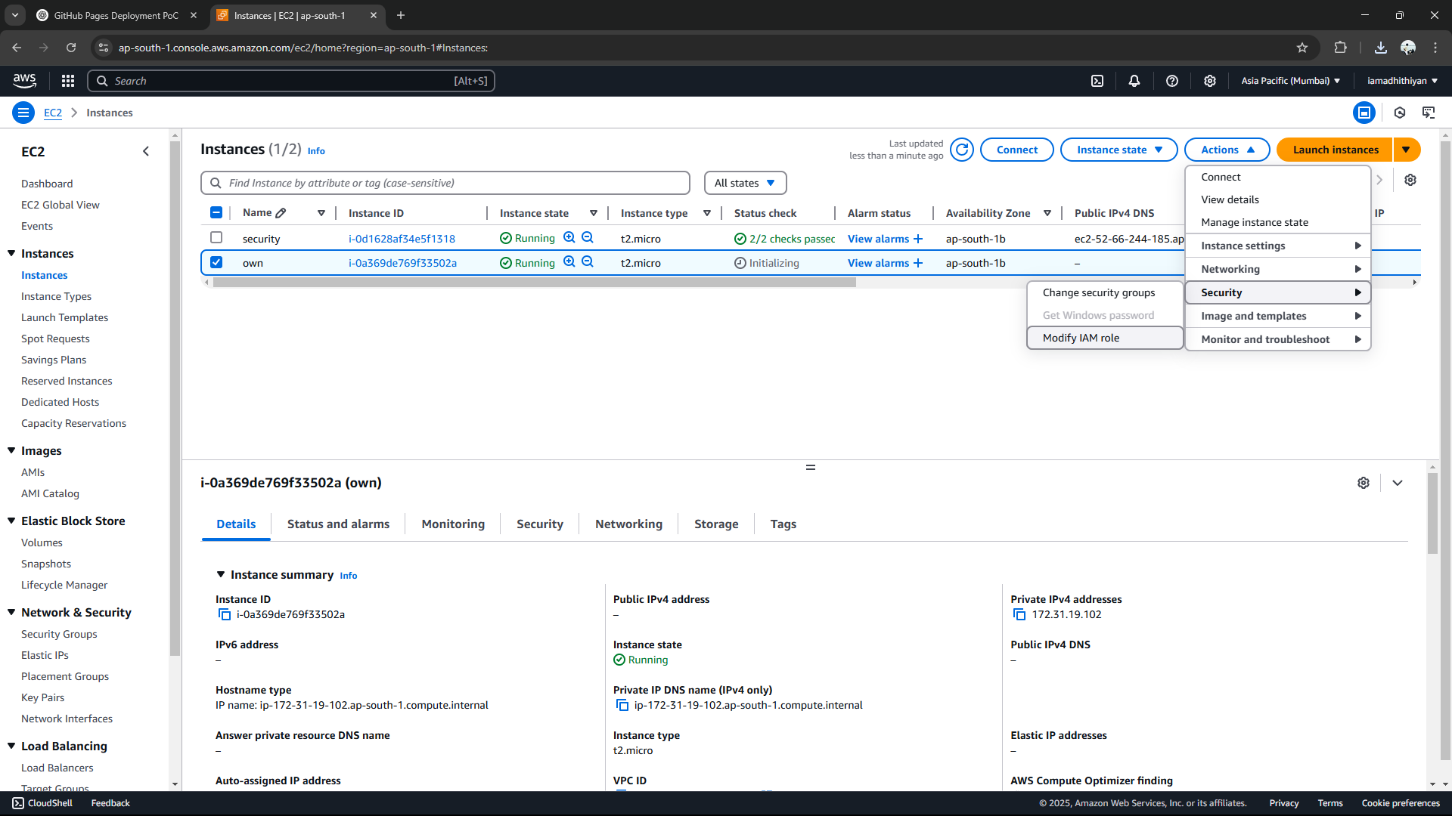
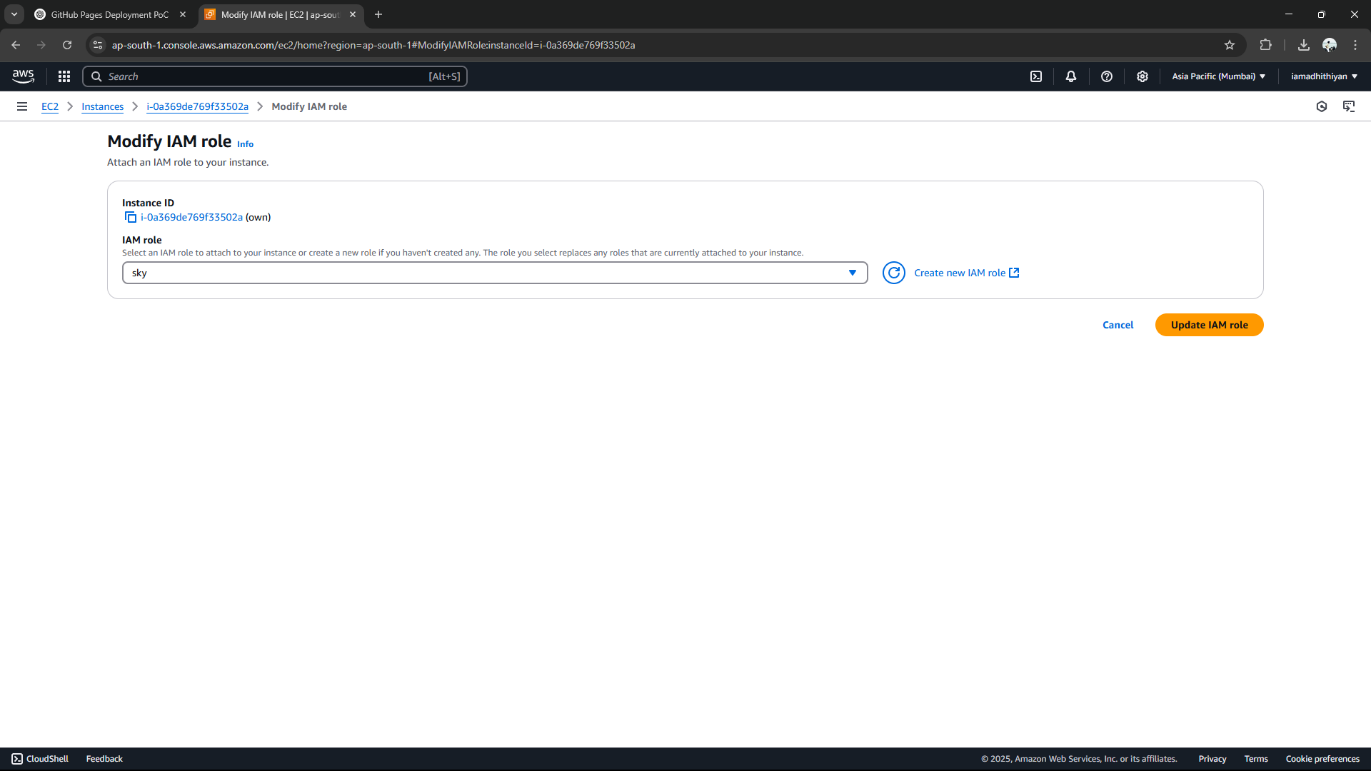
1. **Name & Create the Role**



# Step 2: Attach the Role to Your EC2 Instance

1. **Open the EC2 instance**

1. **Modify IAM Role for the Instance** , Select your **EC2 instance**. Click **Actions → Security →** modify role



# Step 3: Connect to Your Instance via SSH

# Locate the Public IP Address:

# Navigate back to the EC2 Dashboard.

# Under the Instances section, find the instance you launched.

# Note the Public IPv4 address, as you will need it for the SSH connection.

# Adjust Key File Permissions:

# Before using your .pem key file for authentication, set the correct permissions by running:

# chmod 400 /path/to/your-key.pem

# Establish an SSH Connection:

# Open a terminal (or an SSH client like PuTTY if using Windows).

# Use the following command to connect via SSH, replacing <your-key.pem> with the path to your key file and <Public\_IP> with your instance’s public IP address:

# ssh -i your-key.pem ec2-user@<Public\_IP>

# For Windows Users (PuTTY):

# Convert the .pem key to .ppk format using PuTTYgen.

# Open PuTTY, enter the instance's public IP, and go to Connection > SSH > Auth to select your private key.

# Click Open to establish the connection.

# Confirm the SSH Key Fingerprint:

# On the first connection attempt, you’ll receive a prompt to verify the host’s authenticity.

# Type yes to proceed.

# You should now be successfully logged into your EC2 instance!

# Verify IAM Role Permissions:

# Log into the EC2 Instance:

# Connect to the instance via SSH:

# ssh -i your-key.pem ec2-user@<your-instance-public-ip>

# Check the Assigned IAM Role:

# Run the following command to confirm the IAM role associated with the instance:

# Test Access to Amazon S3:

# List available S3 buckets:

# If the IAM role has AmazonS3FullAccess, all S3 buckets will be displayed.

# With AmazonS3ReadOnlyAccess, you can view bucket contents but cannot make modifications.

# To create a new S3 bucket (requires write permissions):

# Validate Restricted Actions:

# If your IAM role lacks write permissions, attempt to delete a bucket:

# aws s3 rb s3://your-new-bucket-name

# If permissions are restricted, you should receive an error message indicating access is denied.

# Top of Form

# Bottom of Form

# Expected Outcome

By completing this POC, you will:

1. **IAM Role Verification:** Running curl on the instance metadata URL should return the IAM role name, confirming it is assigned.

2. **Allowed Actions Succeed:** If permissions are correct, aws s3 ls lists buckets, and aws s3 mb s3://your-new-bucket-name creates one (if write access is enabled).

3. **Unauthorized Actions Fail:** Lacking permissions results in errors, such as an Access Denied message when attempting aws s3 rb s3://your-new-bucket-name.