# 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: Shakthi Lakshmanan V Department: CSE



### ****Introduction****

This Proof of Concept (PoC) explores the implementation of IAM roles and permissions in AWS to enhance security and access control. Instead of relying on hardcoded credentials, IAM roles provide a secure way to manage access to AWS services. This PoC focuses on creating an IAM role, assigning it to an EC2 instance, and verifying its access to services like Amazon S3.

### ****Overview****

The process is structured into key steps:

1. **Creating an IAM Role** – Define a role in AWS IAM and attach policies that grant necessary permissions.
2. **Launching an EC2 Instance** – Set up a virtual machine in AWS for testing IAM role-based access.
3. **Assigning the IAM Role to the EC2 Instance** – Attach the IAM role to allow secure interaction with AWS services.
4. **Verifying Access** – Test permissions by attempting actions on services like Amazon S3 from the EC2 instance.

### ****Objectives****

This PoC aims to:

* **Strengthen Security** – Implement IAM roles to grant temporary, managed access without exposing credentials.
* **Demonstrate Role-Based Access Control** – Enforce permissions through policies that restrict or allow actions.
* **Apply the Principle of Least Privilege** – Ensure only the required permissions are granted, minimizing security risks.
* **Enable Hands-On Learning** – Provide practical experience in managing IAM roles within a cloud environment.

### ****Importance****

IAM roles are critical to secure and efficient cloud operations. They enhance security and governance by:

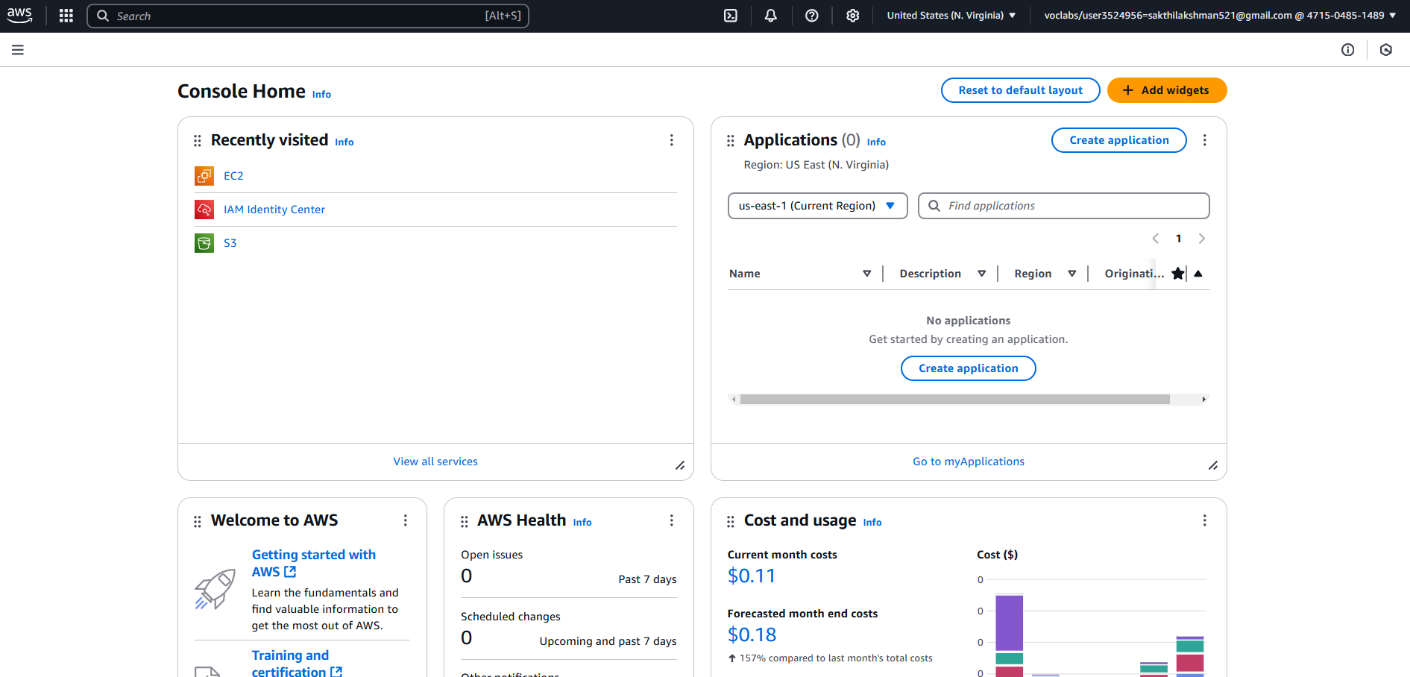
1. **Eliminating Hardcoded Credentials** – Reducing risks by avoiding static access keys in applications or instances.
2. **Enforcing Least Privilege Access** – Restricting permissions to only what is necessary, preventing unauthorized actions.
3. **Ensuring Compliance** – Aligning with security policies, regulatory requirements, and audit best practices.
4. **Enabling Secure Automation** – Allowing AWS resources like EC2 to interact with other services without manual credential management.

By leveraging IAM roles effectively, organizations can enforce strong security measures while ensuring seamless access to cloud resources.

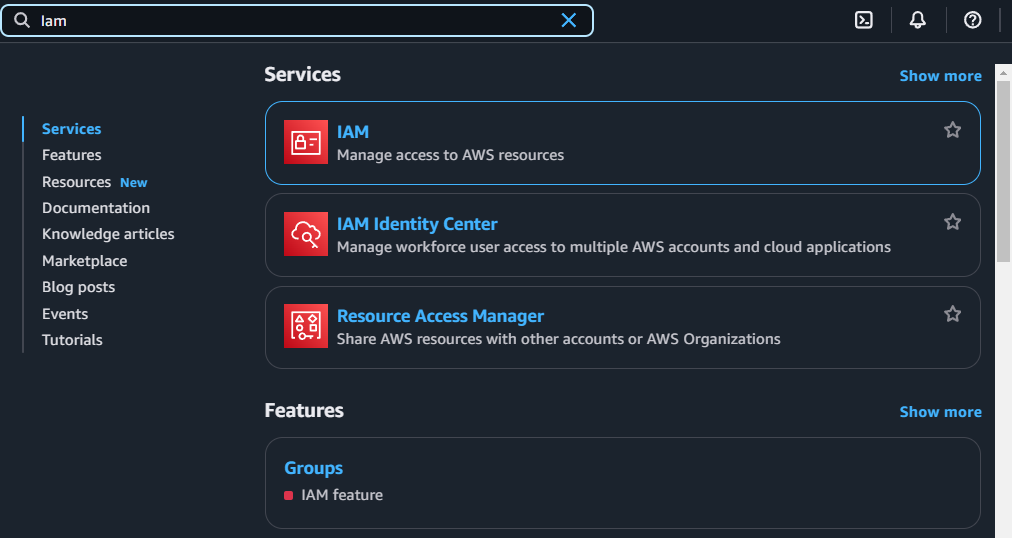
# Step-by-Step Overview

## Step 1:

1. Go to [AWS Management Console](https://aws.amazon.com/console/).
2. Enter your username and password to log in.

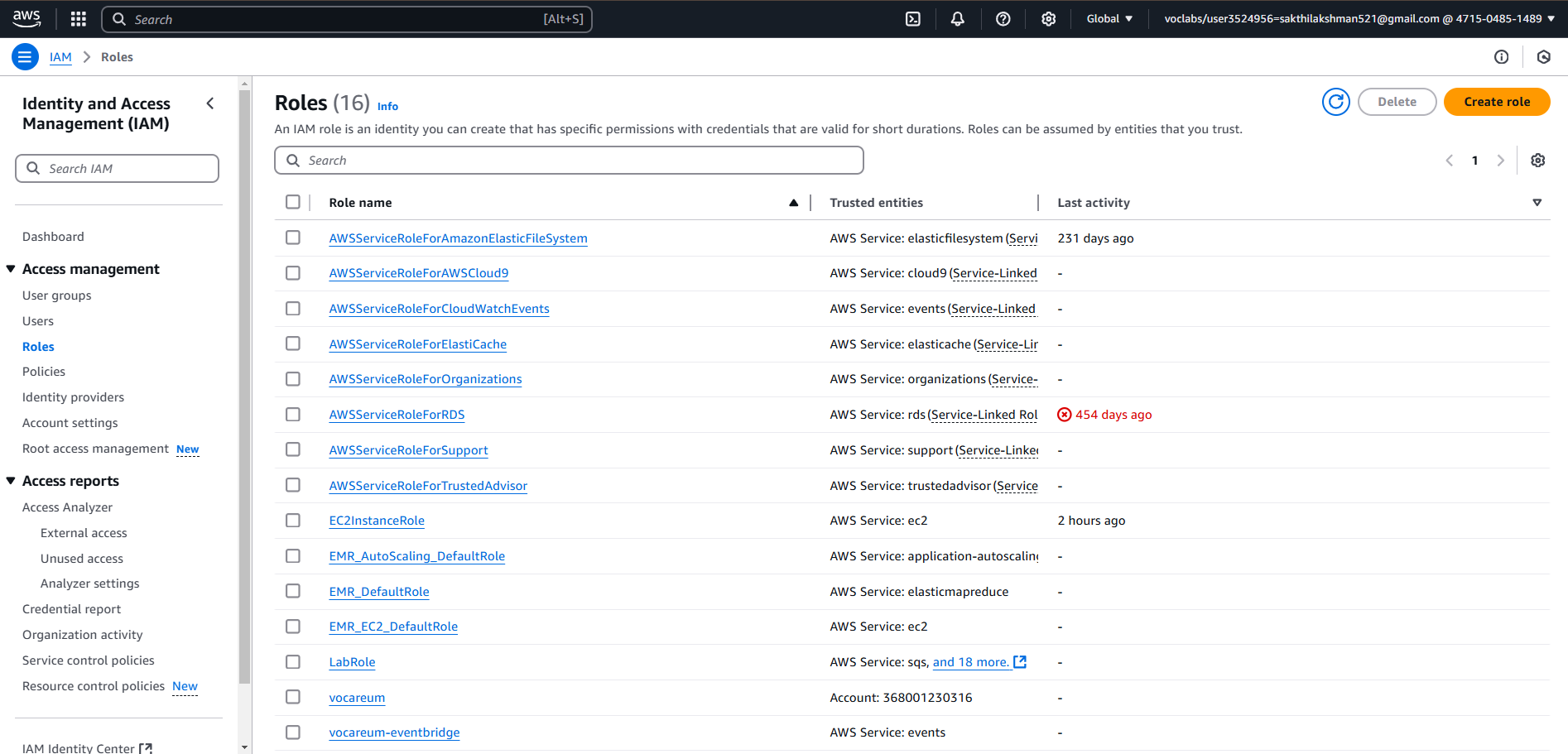


## Step 2:

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

## Step 3:

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

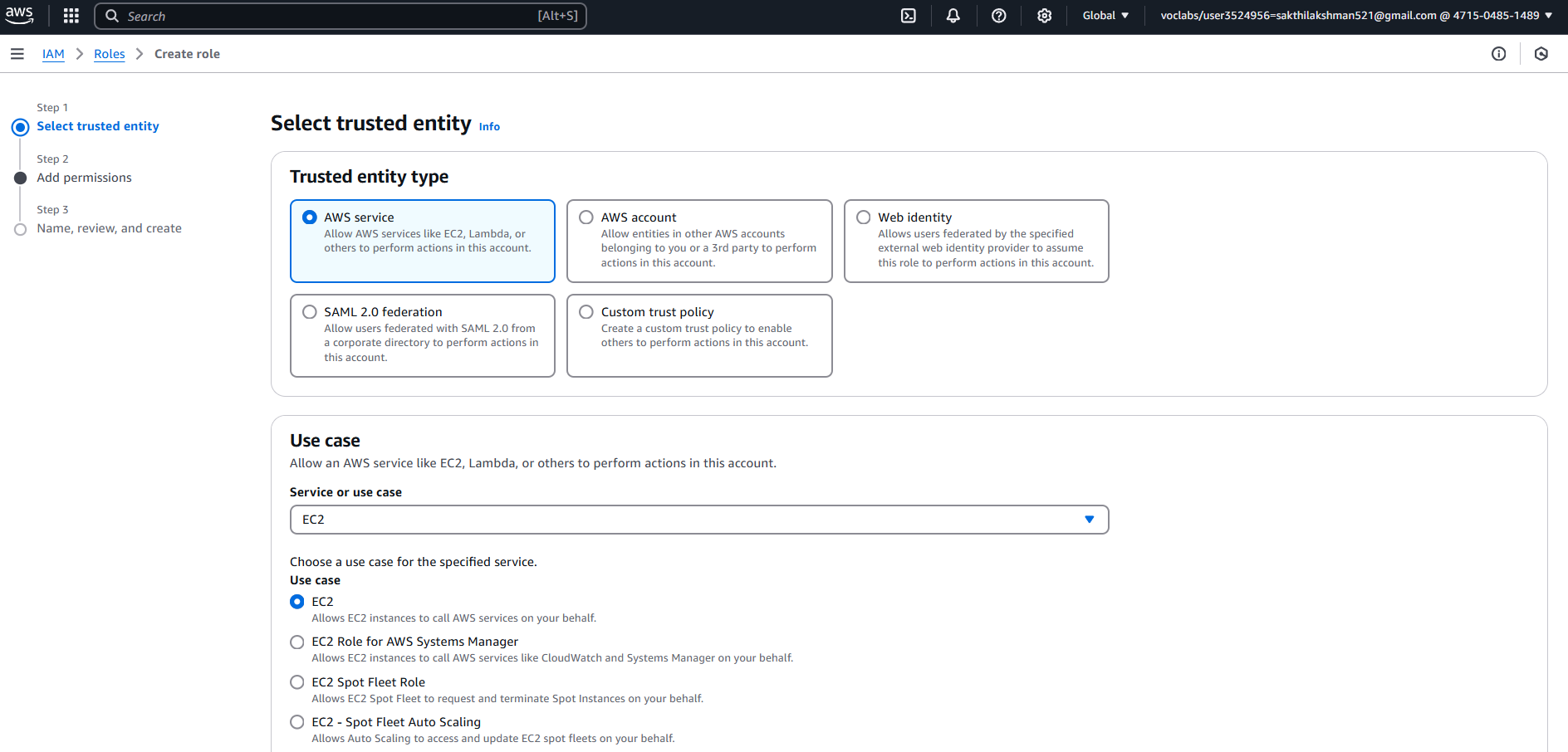


## Step 4:

1. On the **"Create Role"** page, under **Trusted Entity Type**, select

**AWS Service** (it should be selected by default).

1. In the **Use Case** dropdown, choose **EC2**. Click **Next** to continue



## Step 5:

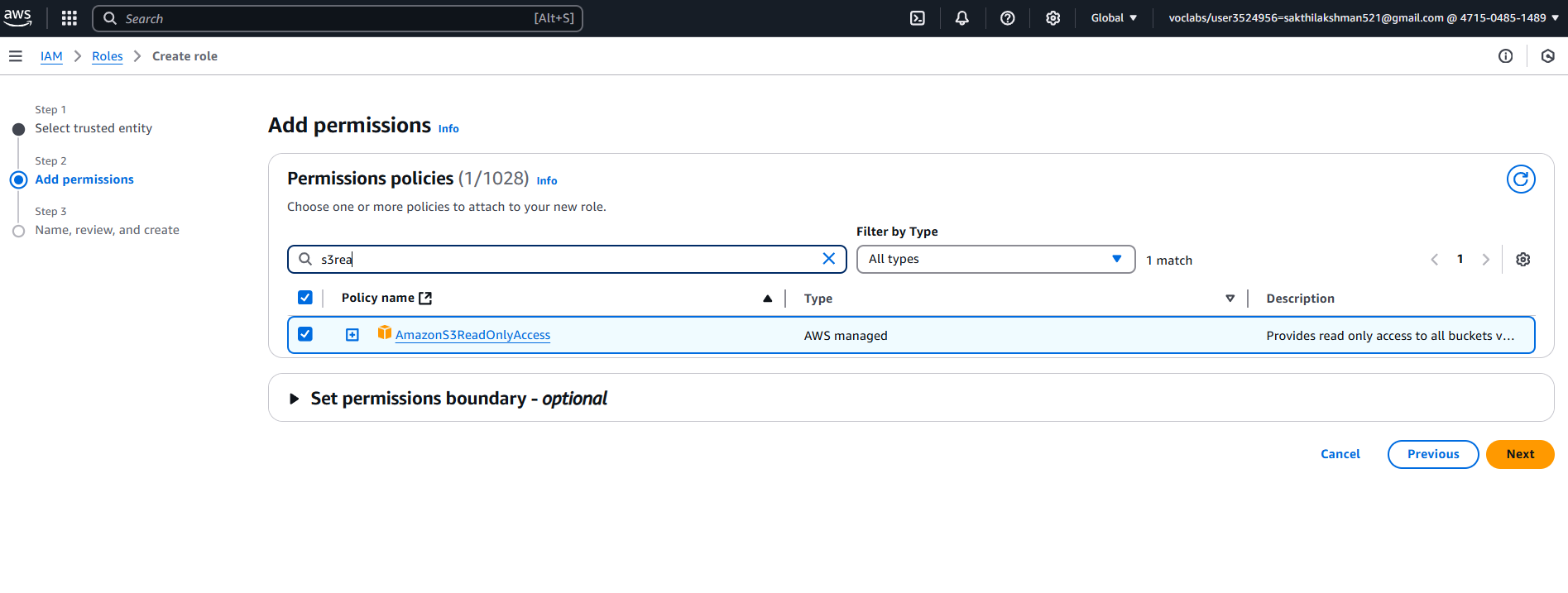
1. On the **Permissions** page, you’ll see a list of policies.
2. Select a policy based on what actions you want the VM to perform. For example:

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

### AmazonS3ReadOnlyAccess.

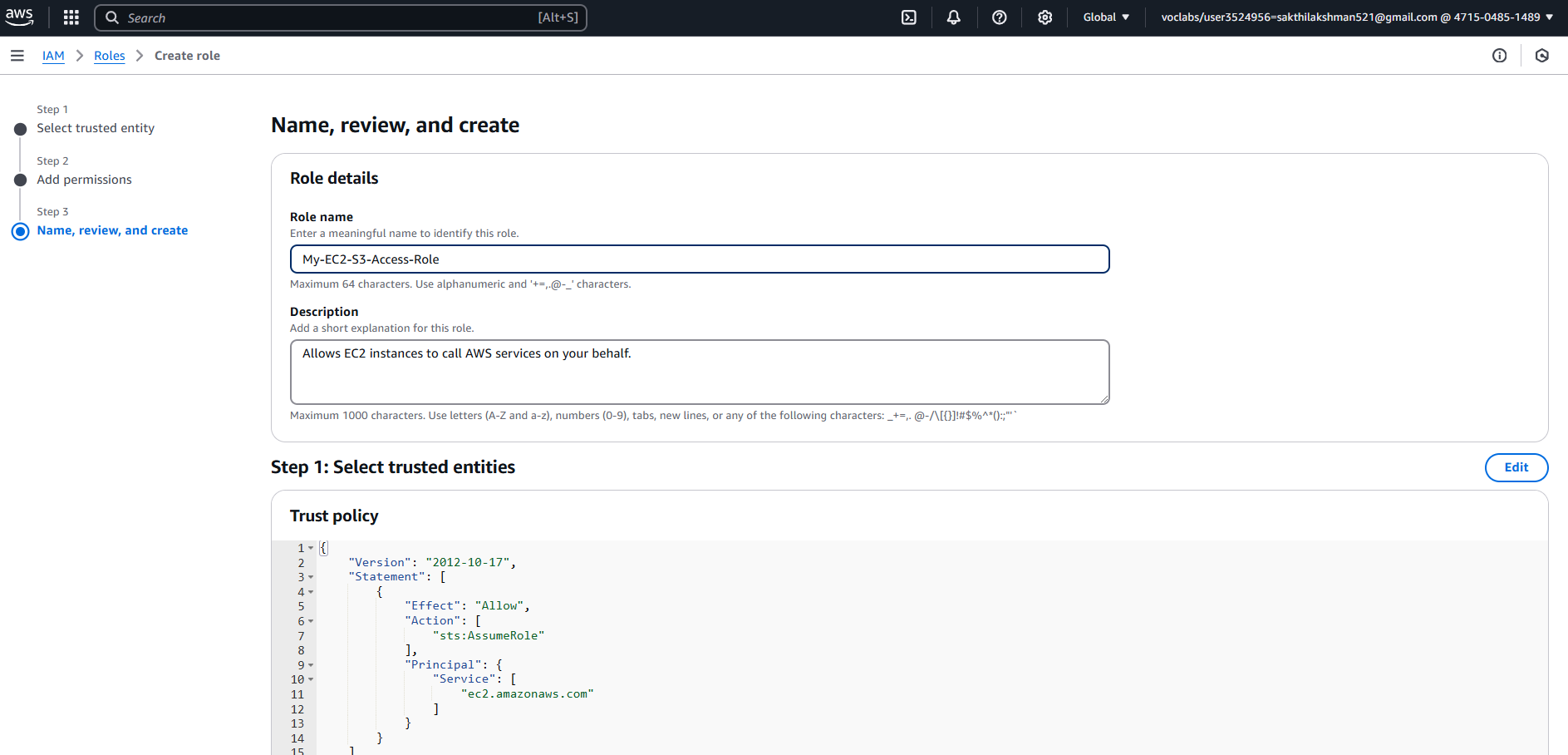
You can search for policies in the search bar (e.g., type "S3" for S3 policies).

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



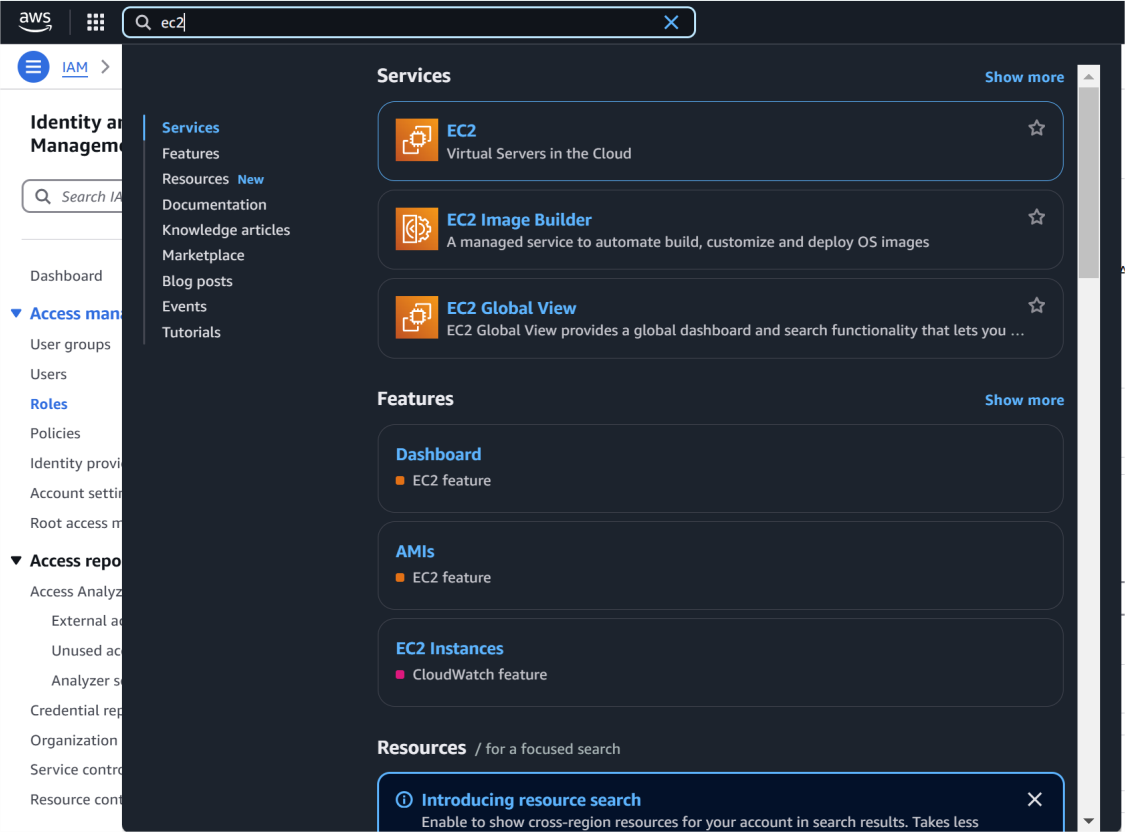
## Step 6:

1. On the **Role Details** page:
   * Enter a name for your role (e.g., S3-Access-Role).
   * (Optional) Add a description or tags if you’d like.
2. Click **Create Role** to finish.



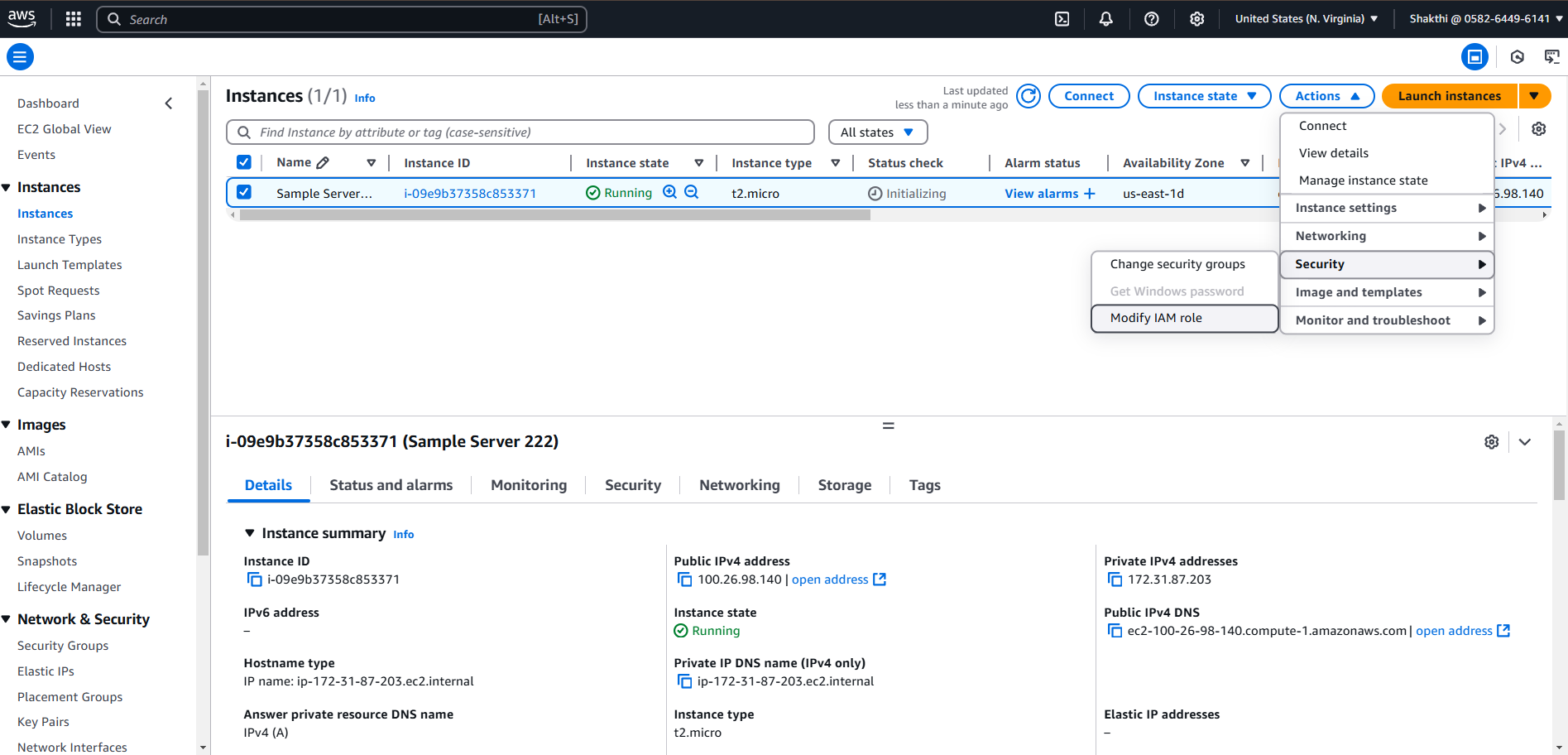
## Step 7:

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 8:

1. In the **Instance details** section, click **Actions** in the top right corner.
2. From the dropdown, choose **Security** > **Modify IAM Role**.

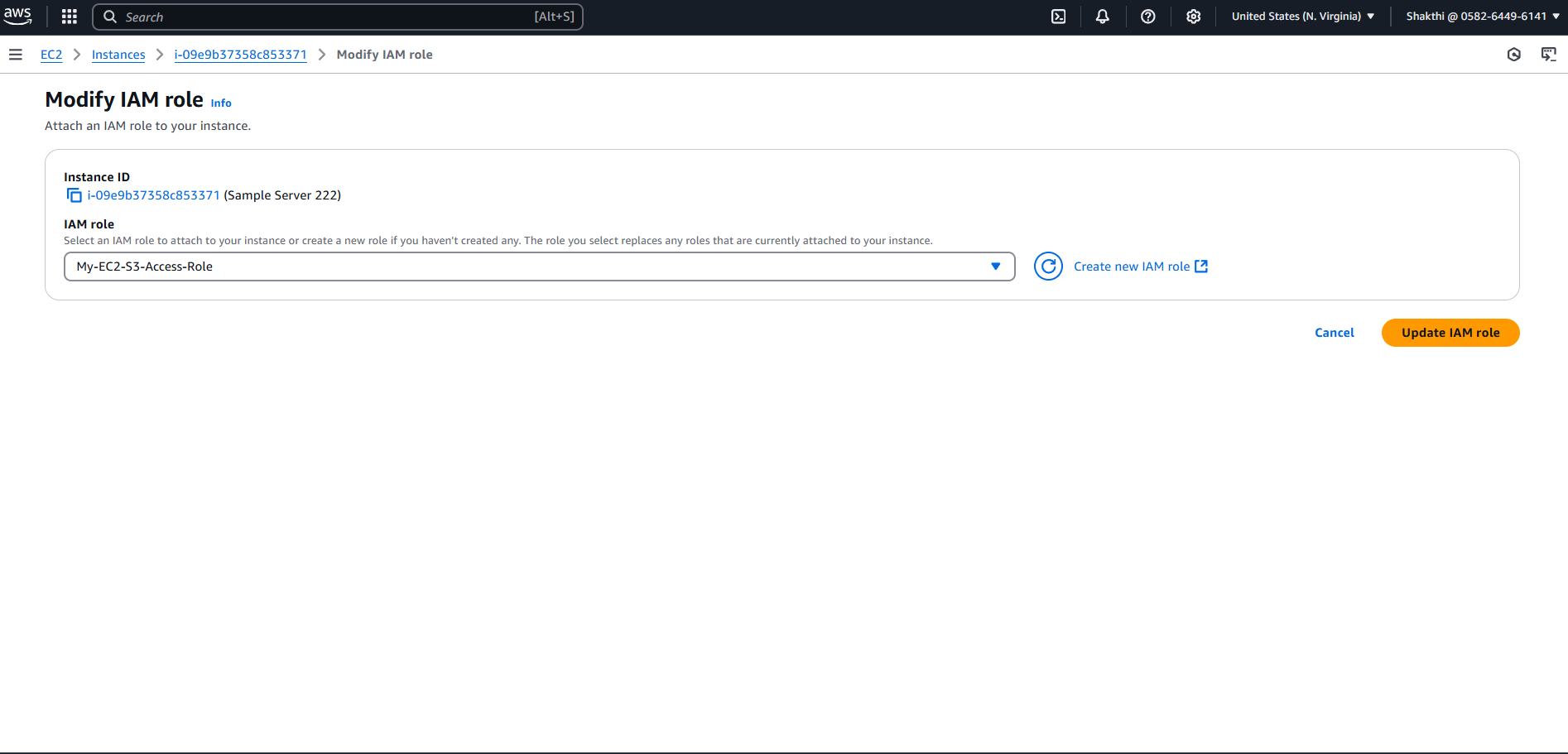


## Step 9:

* 1. In the **Modify IAM role** window, you should see a dropdown for

### IAM role.

* 1. Select the role you created earlier (e.g., S3-Access- Role).
  2. Click **Update IAM role** to apply the changes.



## Step 10:

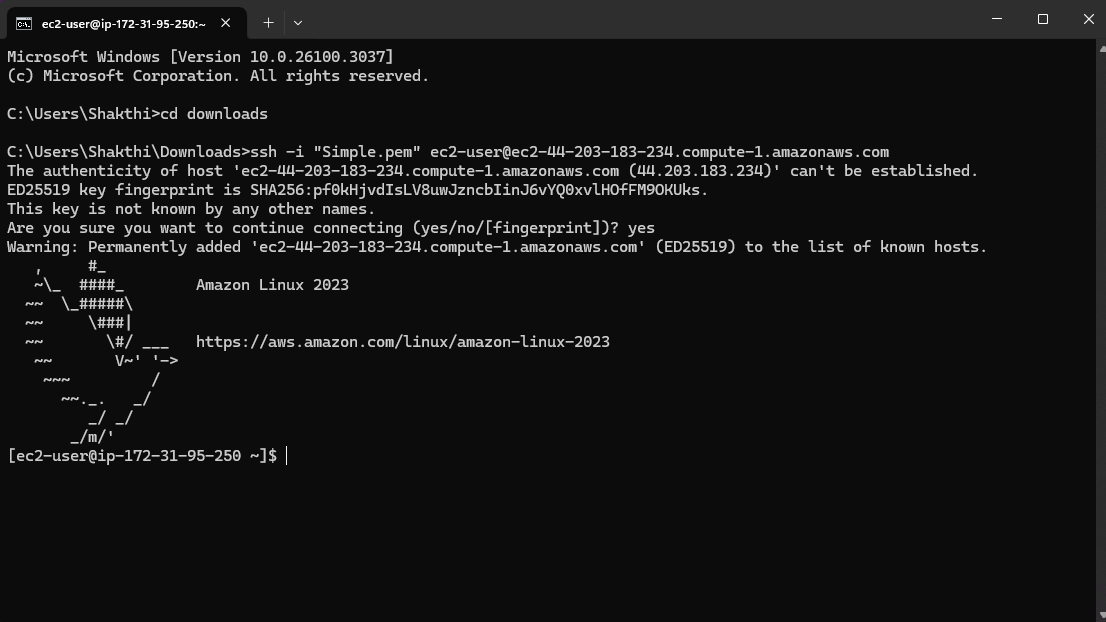
1. Open your terminal (if you're using Linux or macOS) or Command Prompt (Windows).
2. Use SSH to log in to your EC2 instance. For example:

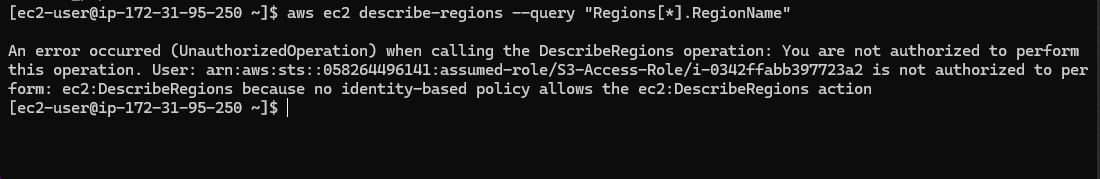
**ssh -i "your-key-pair.pem" ec2-user@your-ec2-public-ip**

## Step 11:

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

The error confirms that your IAM role (My-EC2-S3-Access-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.





# 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.