St. JOSEPH'S GROUP OF INSTITUTIONS
OMR, CHENNAI - 119

aws

Placement Empowerment Program

Cloud Computing and DevOps Centre

create an IAM role with permissions. Assign the role to your VM and verify its effect by attempting permitted/denied actions.

Name: Abdur Rahman M.R Department : CSE

Introduction

In AWS, **IAM (Identity and Access Management)** allows you to securely control access to AWS services. You can create **roles** and **assign permissions** to resources like EC2 instances, ensuring that only authorized actions can be performed by these resources.

In this guide, we will **create an IAM role** that provides **S3 access**, **assign** this role to an **EC2 instance**, and **test the permissions** to ensure it works as expected. This process is crucial for granting your EC2 instances secure access to S3 without the need to hard-code credentials.

Overview

1. **Create an IAM Role** with permissions (e.g., AmazonS3FullAccess).
2. **Assign the IAM Role** to your EC2 instance to allow it to interact with AWS services securely.
3. **Install AWS CLI** on the EC2 instance (if it's not installed) for managing AWS resources.
4. **Test Permissions** by verifying that your EC2 instance can interact with S3.

   By following these steps, you will enable your EC2 instance to access **Amazon S3** (or any other AWS service you configure) securely using the IAM role without needing to manually configure credentials.

Step-by-Step Overview

1. Create the IAM Role for S3 Access

Create Role

Attach Permissions:

Name the Role:

2. Assign IAM Role to EC2 Instance

Modify EC2 Instance:

in your EC2 instance, select it, then click **Actions > Security > Modify IAM Role**.

Attach the Role:

-

Verify Permissions:

```
pcabd@peace MINGW64 ~ (master)
$ cd C:\AWS

pcabd@peace MINGW64 /c/AWS
$ chmod 400 "Aizen.pem"
chmod: changing permissions of 'Aizen.pem': Permission denied

pcabd@peace MINGW64 /c/AWS
$ ec2-54-165-125-54.compute-1.amazonaws.com
bash: ec2-54-165-125-54.compute-1.amazonaws.com: command not found

pcabd@peace MINGW64 /c/AWS
$ ssh -i "Aizen.pem" ec2-user@ec2-54-165-125-54.compute-1.amazonaws.com
The authenticity of host 'ec2-54-165-125-54.compute-1.amazonaws.com (54.165.125.
54)' can't be established.
ED25519 key fingerprint is SHA256:rzaWnl5bXD/W1YL9OaUJli9MhaM6LFsIx/9hXCNUbJw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-165-125-54.compute-1.amazonaws.com' (ED25519)
 to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-80-175 ~]$ aws s3 ls
2025-02-02 08:54:56 aws-cloudtrail-logs-314146319737-03e1acf3
2025-02-02 08:52:21 aws-cloudtrail-logs-314146319737-09b82146
2025-02-02 10:07:04 aws-cloudtrail-logs-314146319737-2a43b09b
2025-02-02 10:09:34 aws-cloudtrail-logs-314146319737-ed81cf1a
[ec2-user@ip-172-31-80-175 ~]$
```

SSH into the EC2 Instance:

Test S3 Access:

aws s3 ls

Expected Outcome

Successful S3 Access: You'll see a list of your S3 buckets when you run aws s3 ls.

Access Denied: If you attempt actions outside of your permissions, you'll receive an **AccessDenied** error.