**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

Host a Static Website on a Cloud VM Install Apache on your cloud VM and host a simple HTML website.

Name: Mathimalar.P Department : ADS



**Introduction and Overview**..

IAM (Identity and Access Management) roles in cloud platforms like AWS allow users to control permissions for virtual machines (VMs) and other cloud resources. Instead of using static credentials, IAM roles provide temporary, dynamically managed permissions that help enhance security and streamline access management. Assigning roles to VMs ensures they can perform only the necessary actions, such as accessing specific AWS services or restricting certain operations.

**Objective**

The objective is to create and configure an IAM role, then assign it to a virtual machine (EC2 instance) to define what actions it can perform. This includes:

* Creating an IAM role with required permissions.
* Attaching the role to a VM (EC2 instance).
* Restricting or allowing access to cloud services like S3, DynamoDB, or CloudWatch.
* Managing security by avoiding the use of long-term credentials.

Importance

 **Enhanced Security** – IAM roles reduce reliance on static credentials, minimizing security risks.

 **Fine-Grained Access Control** – Permissions can be precisely defined to allow only necessary actions.

 **Automated Access Management** – Roles can be assigned dynamically without manual credential handling.

 **Compliance and Best Practices** – Helps organizations follow cloud security best practices.

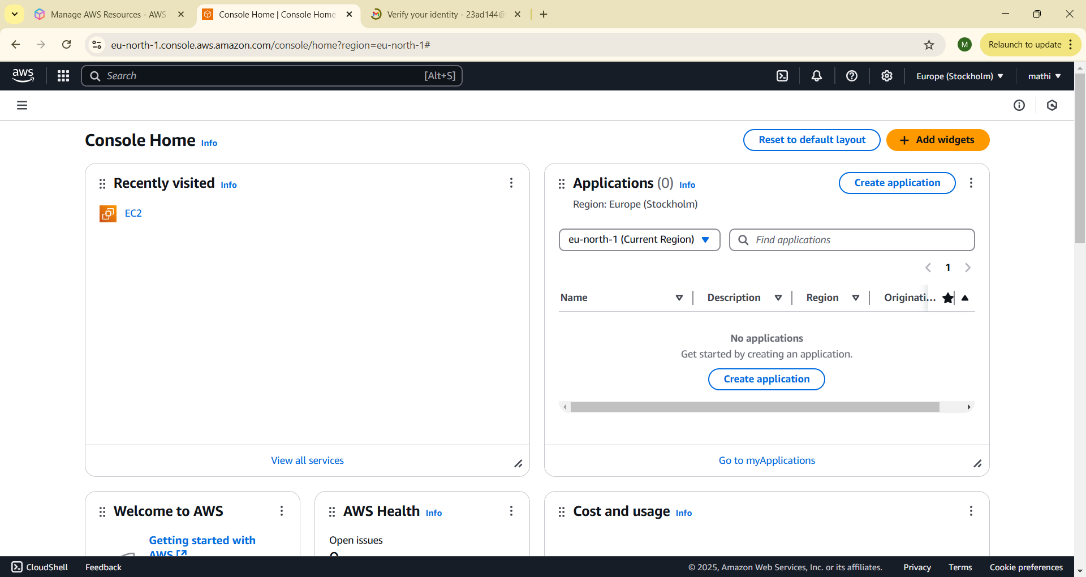
 **Scalability** – Easily assign roles to multiple VMs without manual intervention.

**Step-by-Step Overview**

Step 1:

1. Go to AWS Management Console.

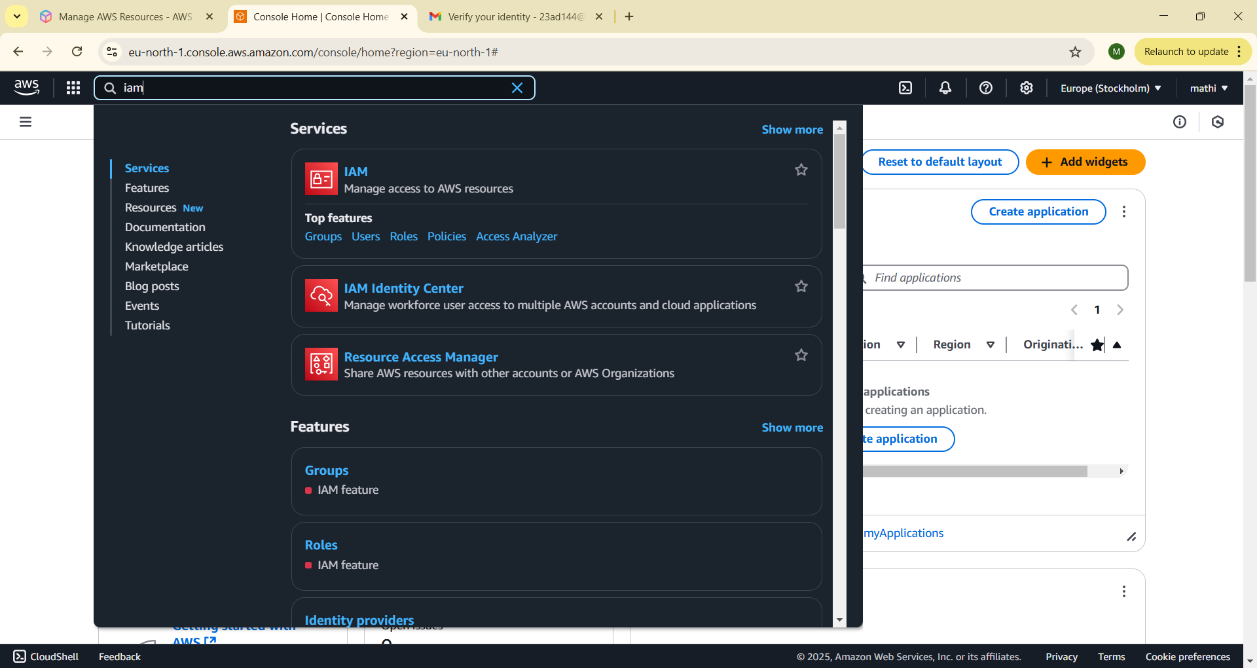
2. Enter your username and password to log in.



Step 2:

1. In the AWS Management Console, type "IAM" in the search bar at the top.

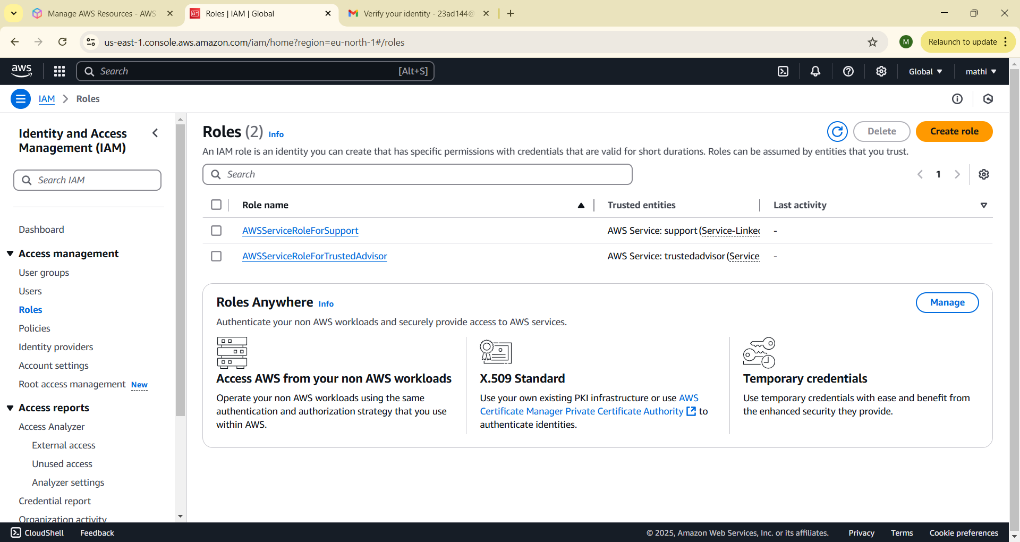
2. Click on IAM from the search results.



Step 3:

1. On the IAM dashboard, click on "Roles" in the left-hand menu.

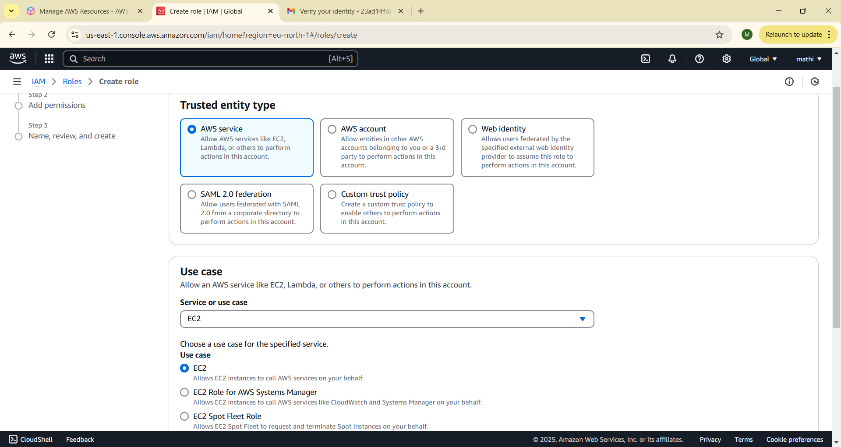
2. On the Roles page, click the "Create Role" button.



Step 4:

1. On the "Create Role" page, under Trusted Entity Type, select AWS Service (it should be selected by default).

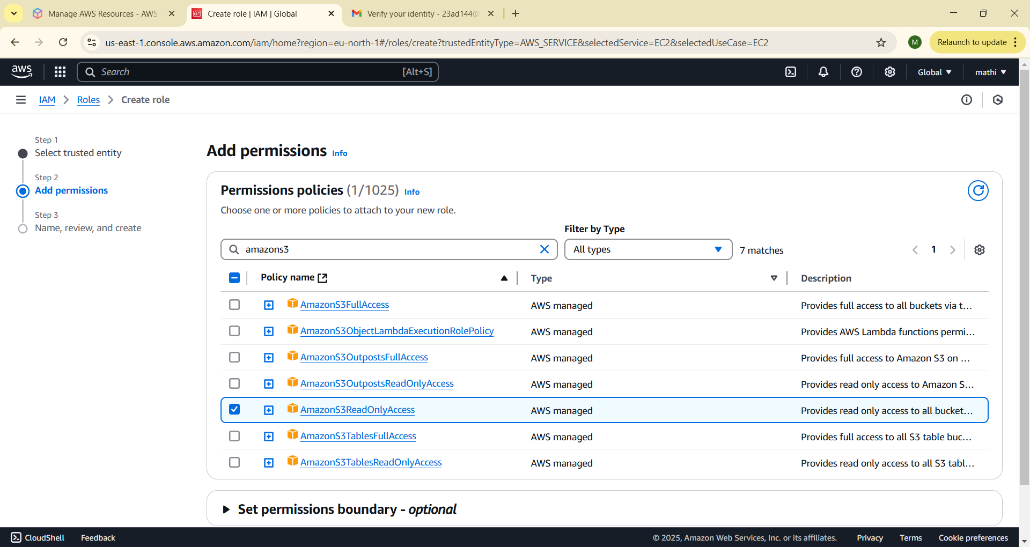
2. In the Use Case dropdown, choose EC2. Click Next to continue



Step 5: 1. On the Permissions page, you’ll see a list of policies.

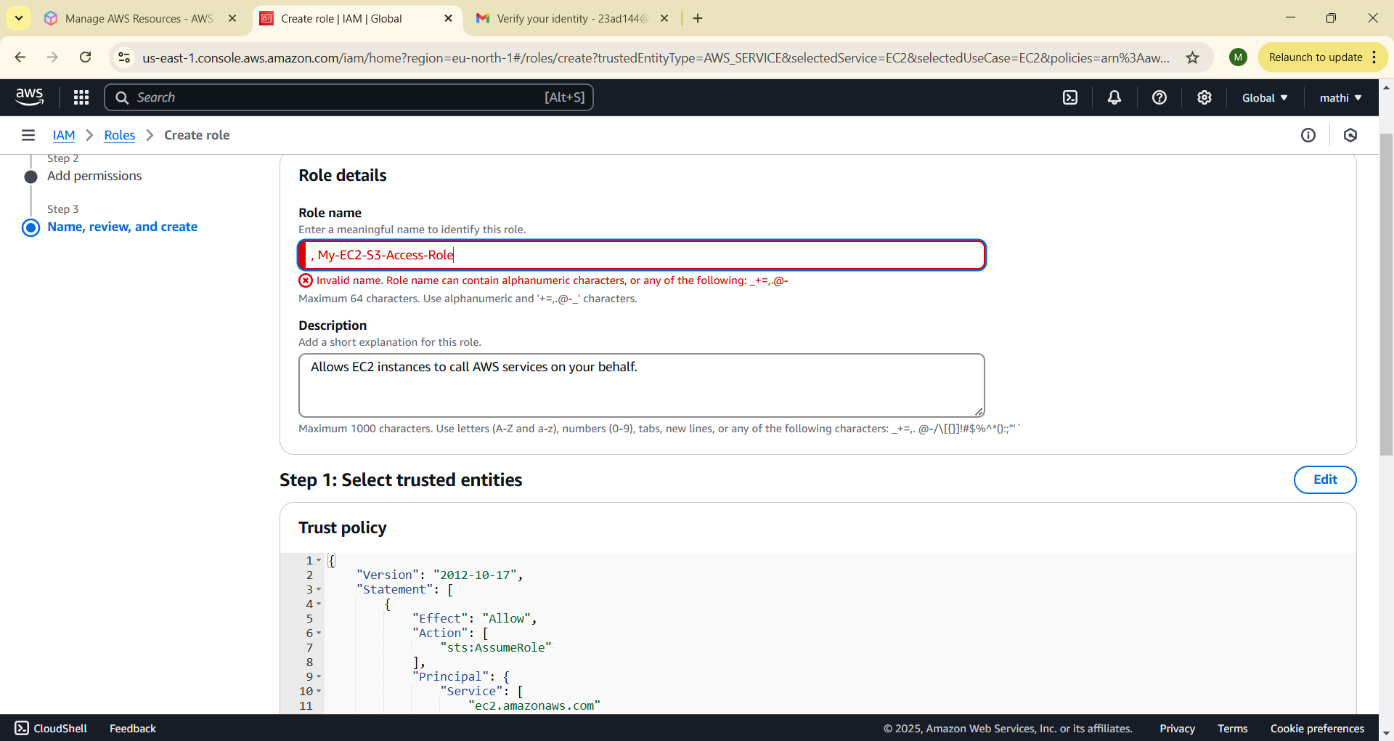
2. Select a policy based on what actions you want the VM to perform. For example: To give the VM read-only access to S3, select AmazonS3ReadOnlyAccess. You can search for policies in the search bar (e.g., type "S3" for S3 policies).

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



Step 6:

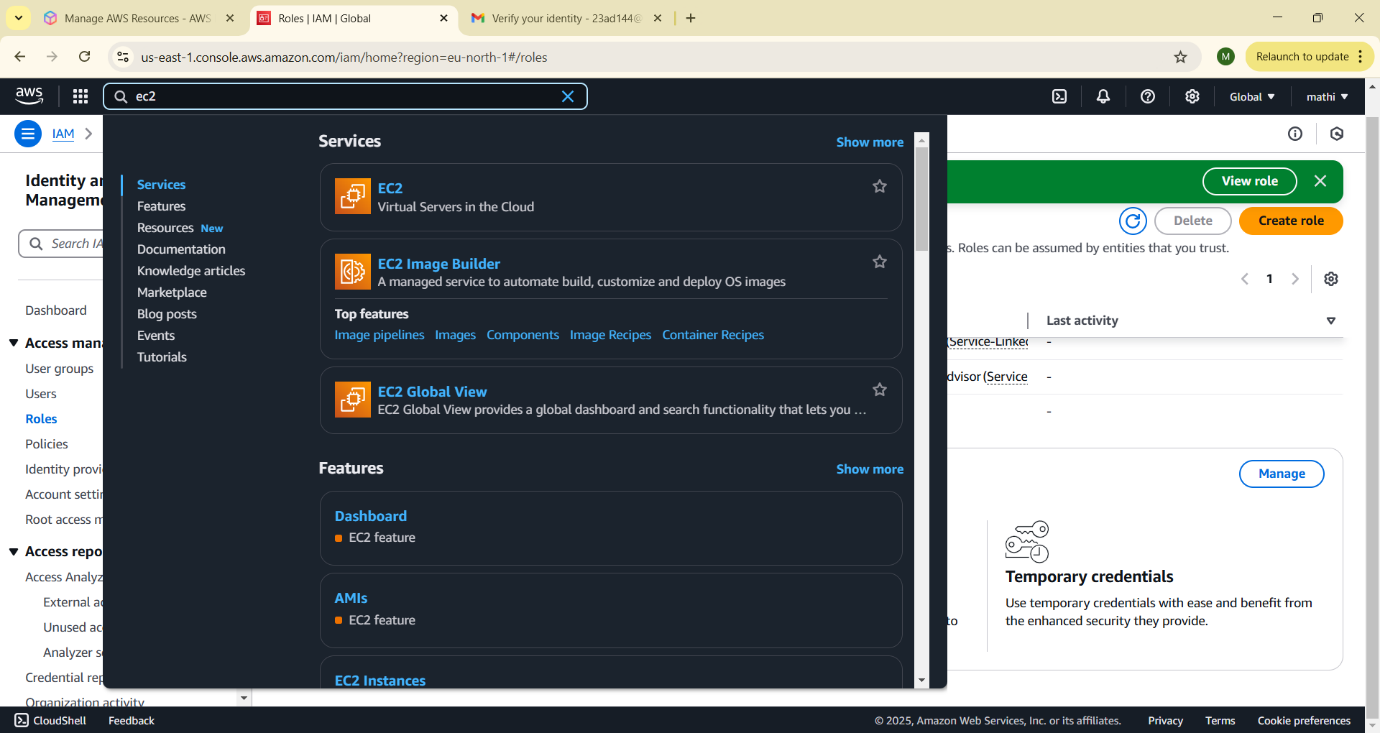
1. On the Role Details page: • Enter a name for your role (e.g., My-EC2-S3-Access-Role). • (Optional) Add a description or tags if you’d like.
2. Click Create Role to finish.



Step 7:

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

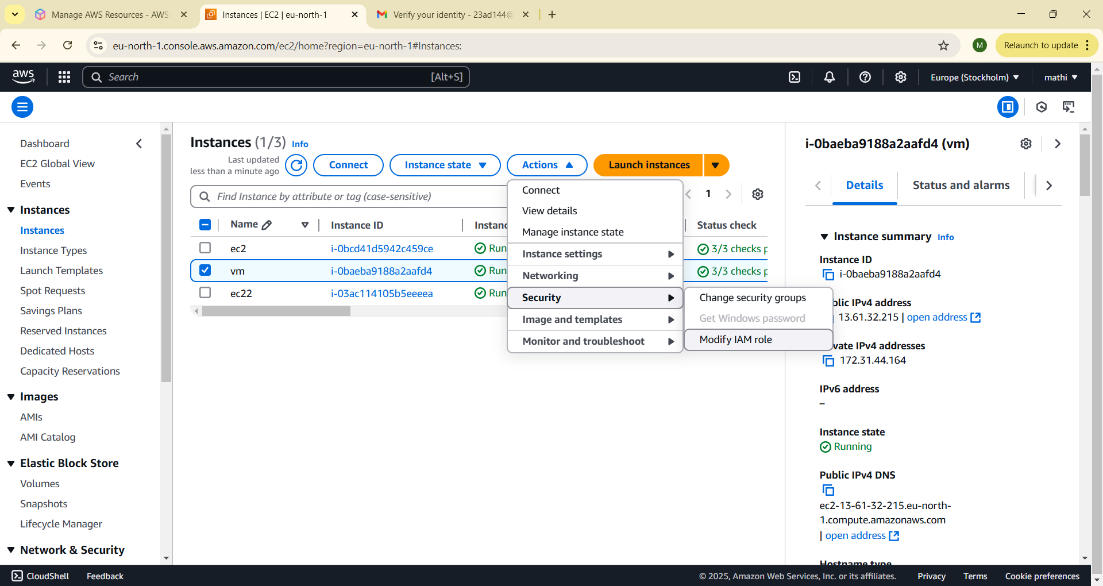
2. Select the instance (VM) you want to assign the IAM role to.



Step 8:

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

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

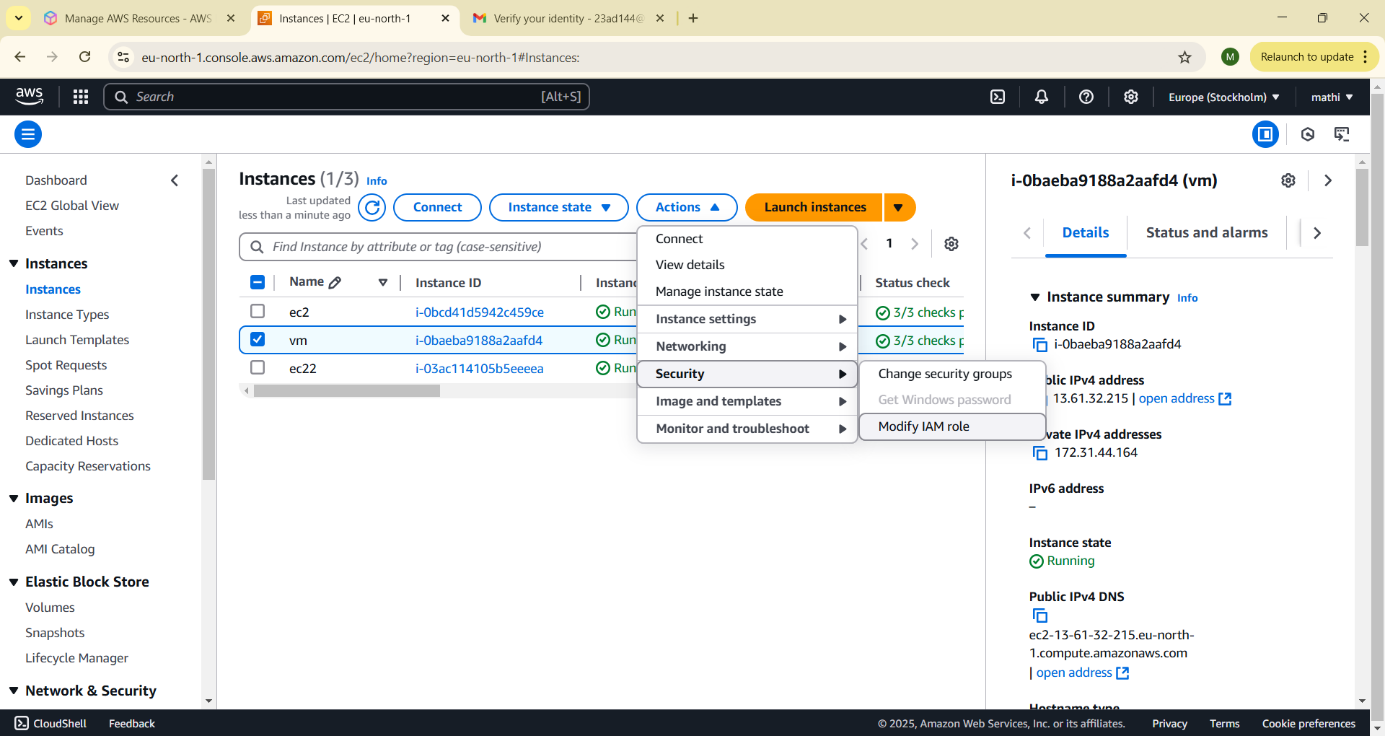


Step 9:

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

2. Select the role you created earlier (e.g., My-EC2-S3-AccessRole).

3. 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.

**Expected Outcome**

 A properly configured IAM role assigned to a virtual machine.

 The VM will have only the required permissions to interact with AWS services.

 Secure and controlled access management without hardcoded credentials.

 Improved cloud security posture and operational efficiency.