AMITY SCHOOL OF ENGINERRING & TECHNOLOGY

(Academic Year 2022-23)

LAB- 5

EC2 Instances

Student Name: Vikas Rajbhar

Class: B. Tech CSE Semester: 7

Enrolment Number: A70405219037

Faculty In-charge
{Department of CSE}
ASET, AUM

AIM OF THE EXPERIMENT:

Perform the following tasks:

1. Create an EC2 Instance
2. Demonstrate the working of static and dynamic website through EC2
3. Shift files from S3 to EC2
4. Shift files from EC2 to S3

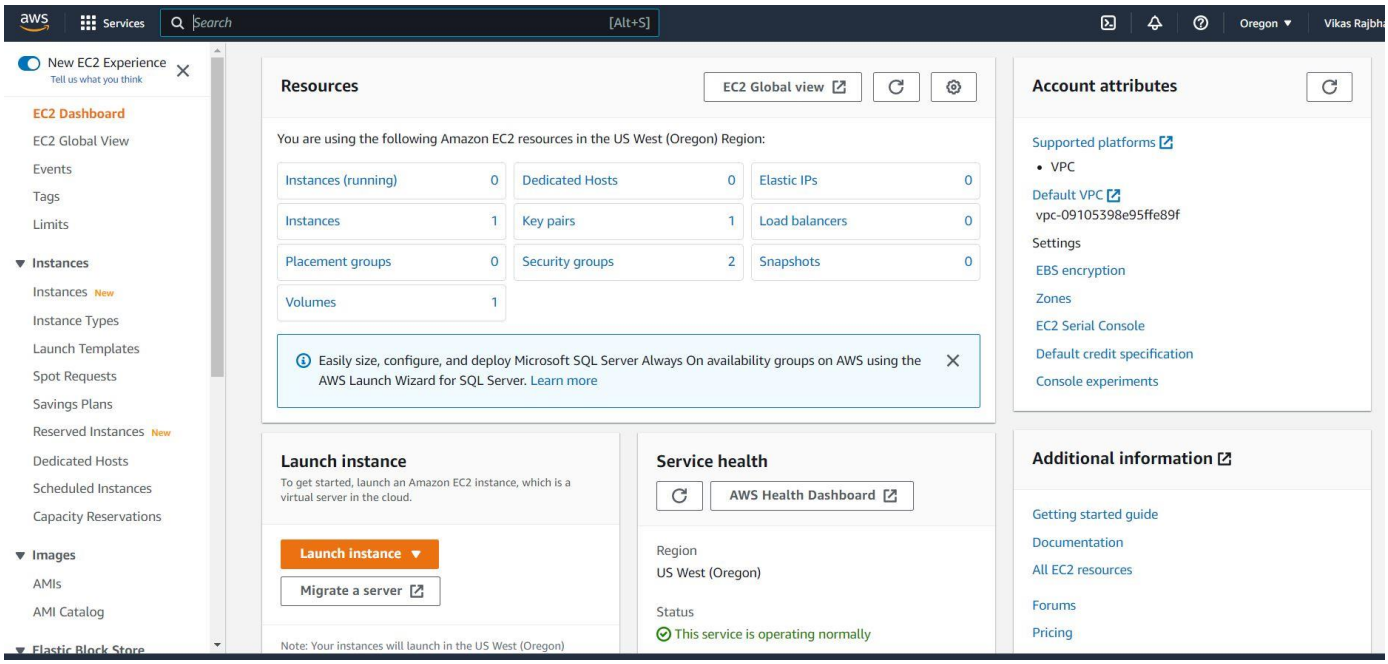
THEORY:

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.

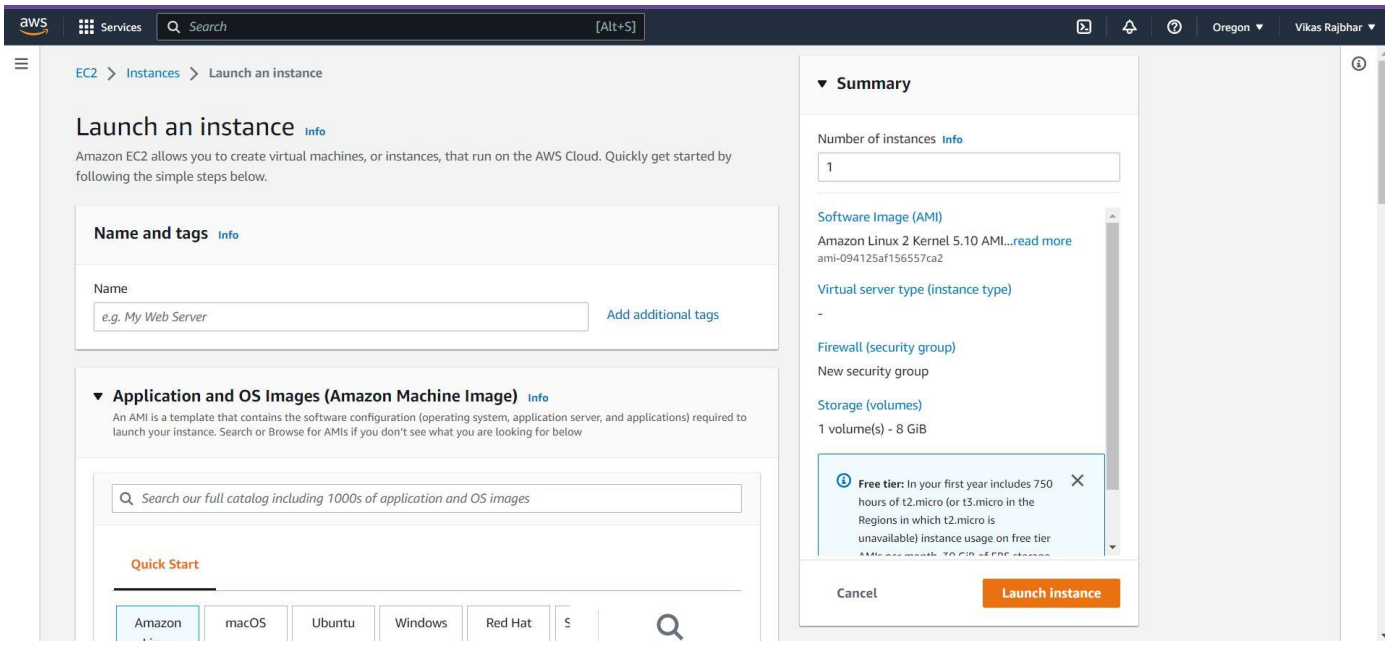
Creating an EC2 Instance

Launching Amazon EC2 Linux Instances:

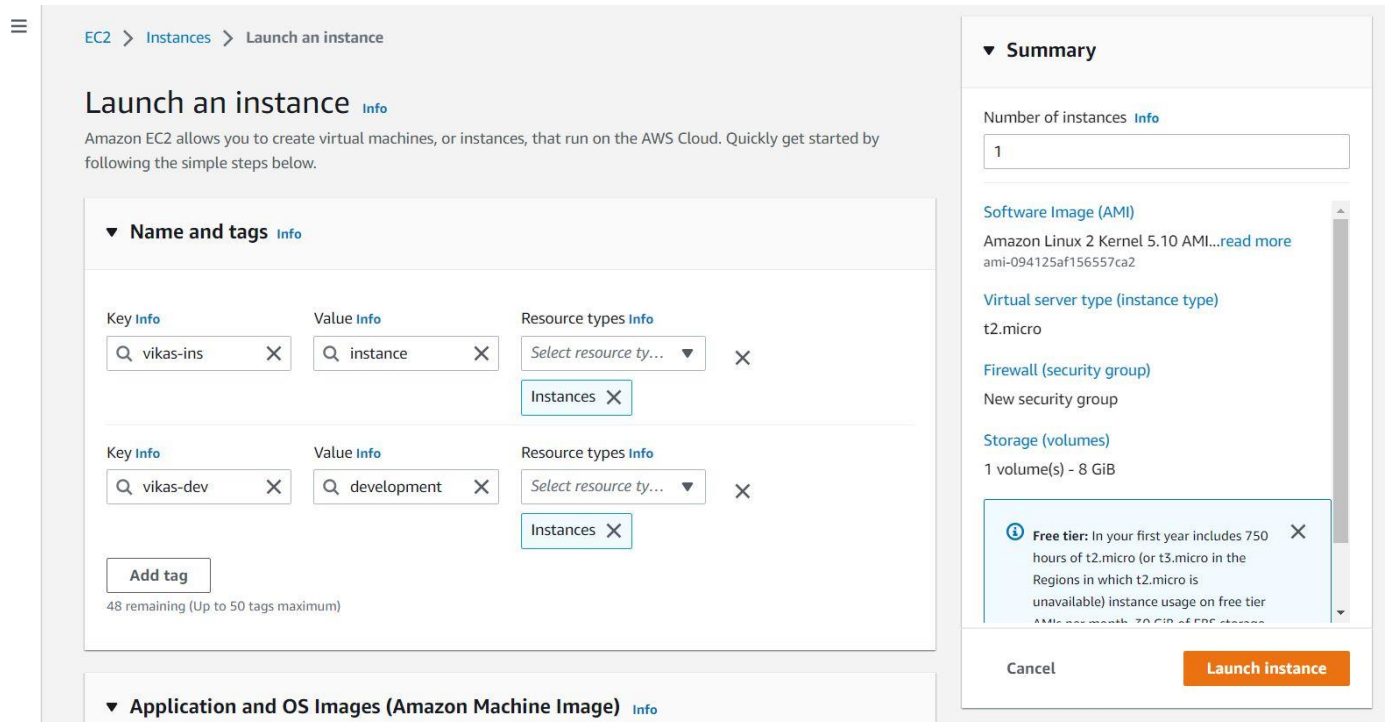
1. Open the Amazon EC2 console.



2. From the EC2 console dashboard, in the **Launch instance** box, choose **Launch instance**, and then choose **Launch instance** from the options that appear.



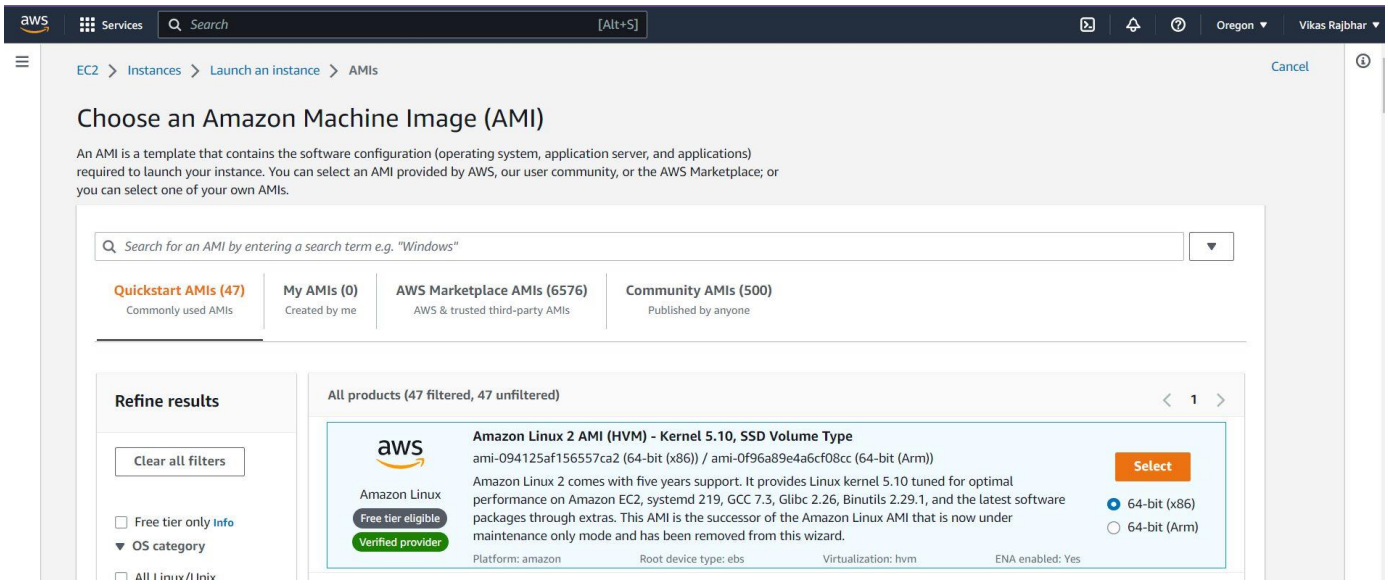
3. Under **Name and tags**, for **Name**, enter a descriptive name for your instance.

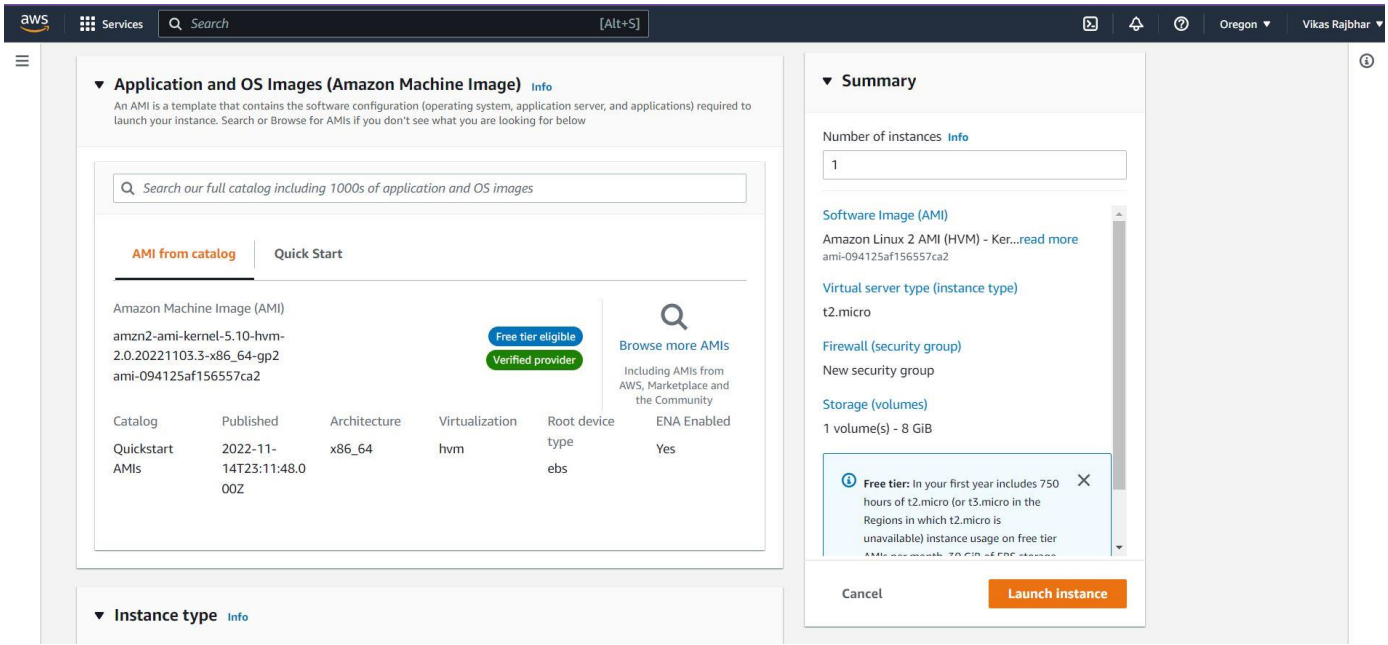


4. Under **Application and OS Images (Amazon Machine Image)**, do the following:

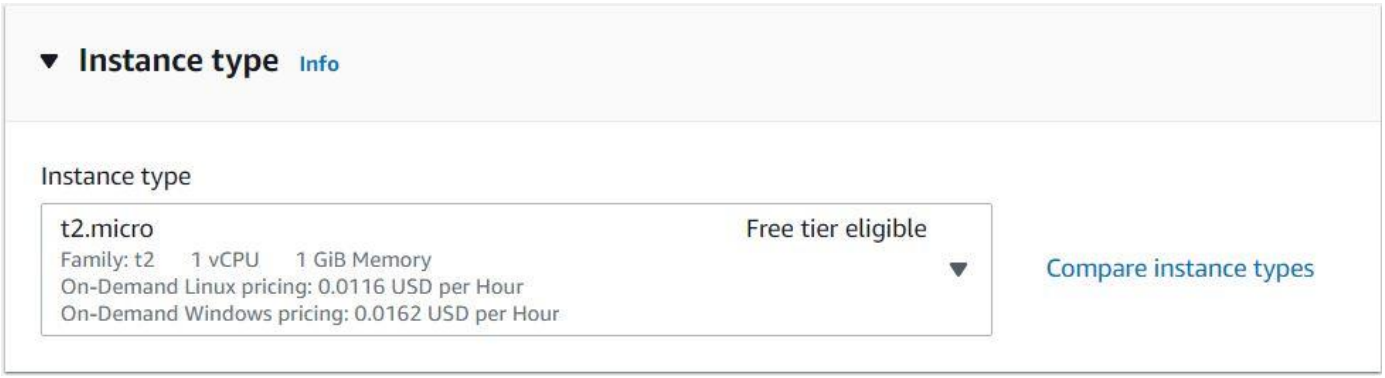
   a. Choose **Quick Start**, and then choose Amazon Linux. This is the operating system (OS) for your instance.

   b. From **Amazon Machine Image (AMI)**, select an HVM version of Amazon Linux 2. Notice that these AMIs are marked **Free tier eligible**. An *Amazon Machine Image (AMI)* is a basic configuration that serves as a template for your instance.

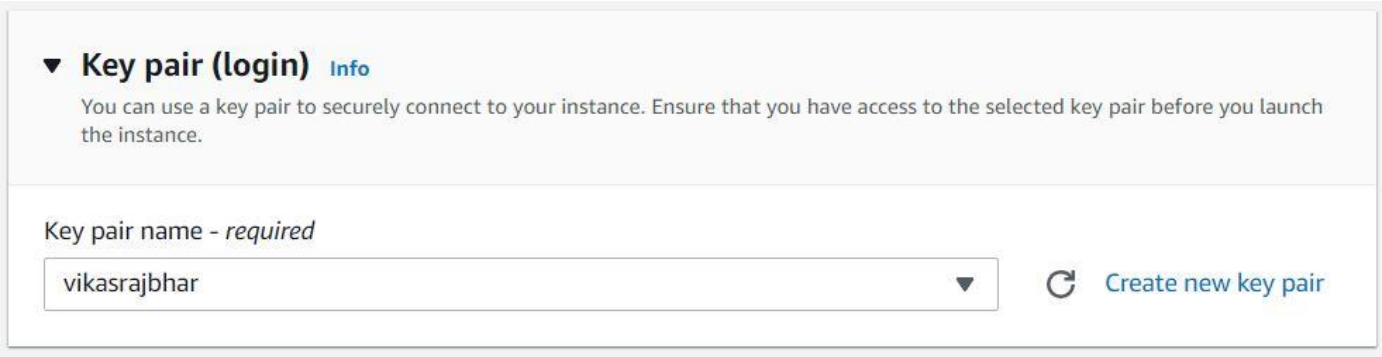




5. Under **Instance type**, from the **Instance type** list, you can select the hardware configuration for your instance. Choose the t2.micro instance type, which is selected by default.



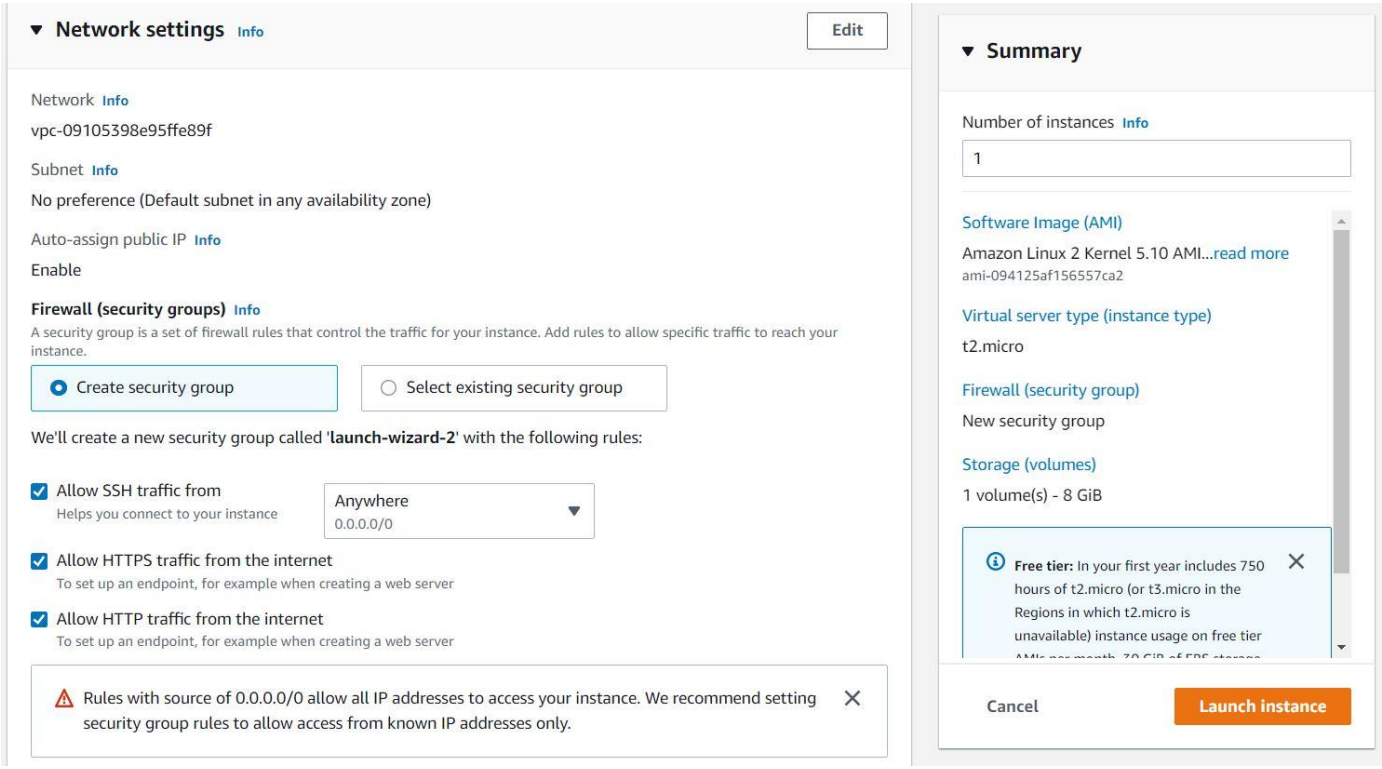
6. Under **Key pair (login)**, for **Key pair name**, choose the key pair that you created when getting set up.



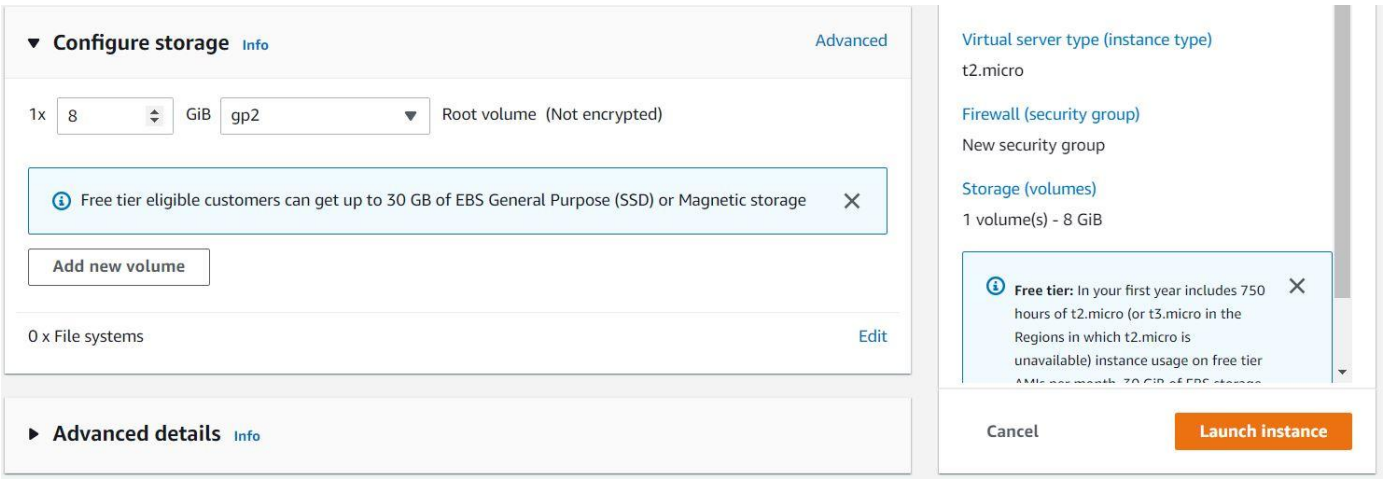
7. Next to **Network settings**, choose **Edit**. For **Security group name**, you'll see that the wizard created and selected a security group for you. You can use this security group, or alternatively you can select the security group that you created when getting set up using the following steps:

   a. Choose **Select existing security group**.

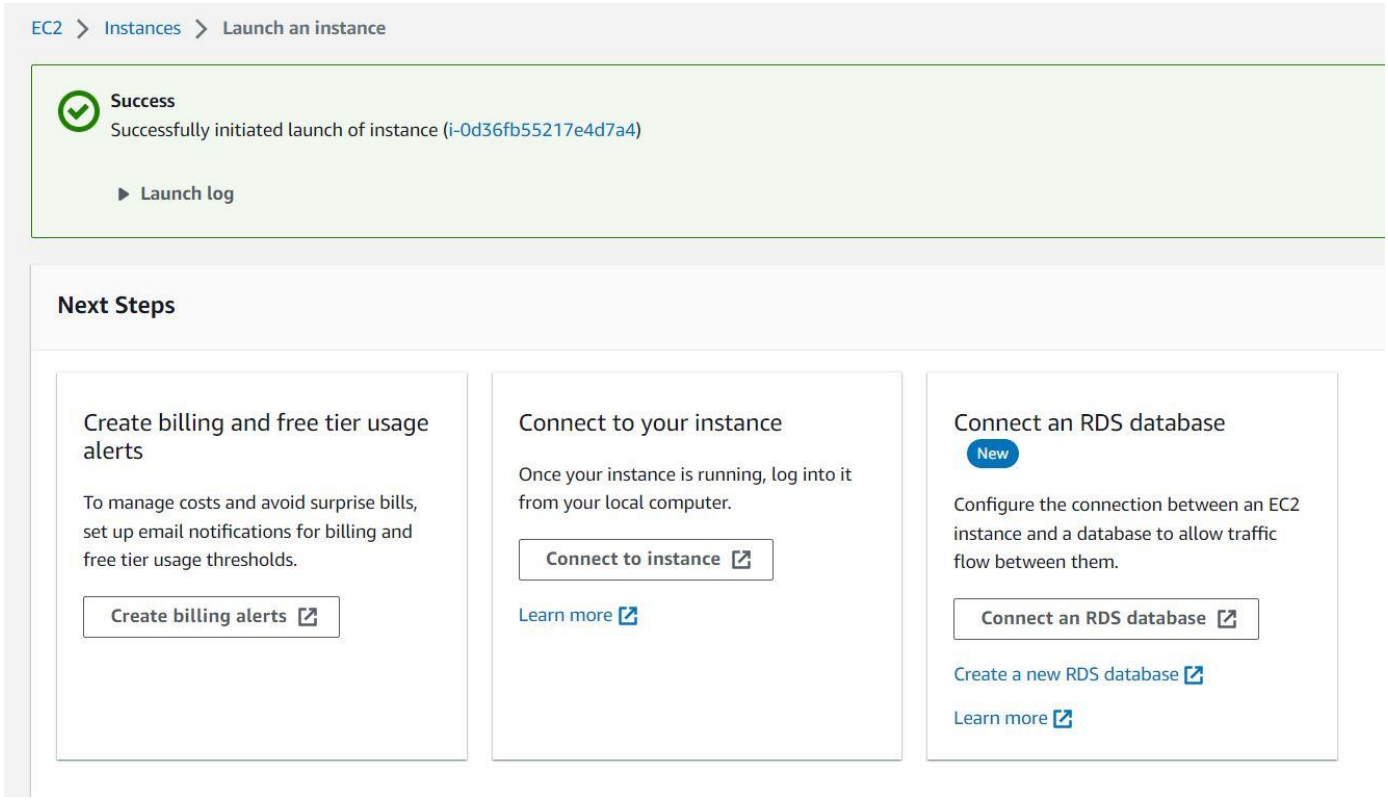
   b. From **Common security groups**, choose your security group from the list of existing security groups.



8. Keep the default selections for the other configuration settings for your instance.
9. Review a summary of your instance configuration in the **Summary** panel, and when you're ready, choose **Launch instance**.

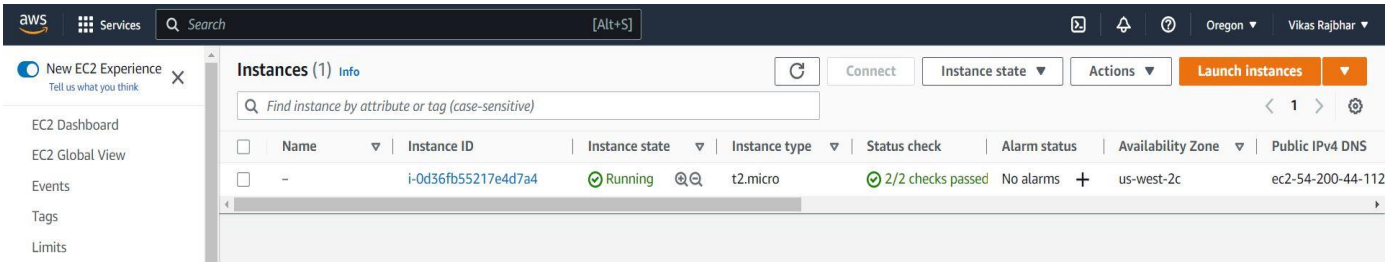


10. A confirmation page lets you know that your instance is launching. Choose **View all instances** to close the confirmation page and return to the console.



11. On the **Instances** screen, you can view the status of the launch. After the instance starts, its state changes to running and it receives a public DNS name.

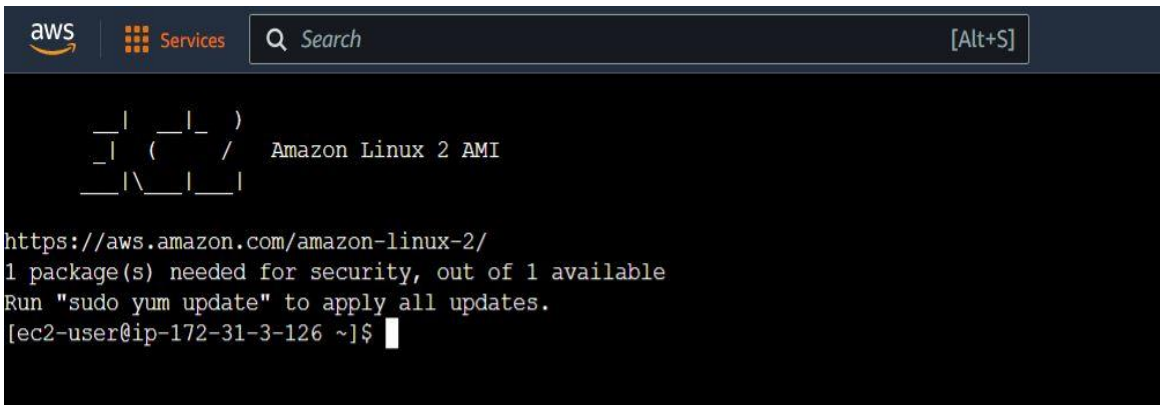
12. It can take a few minutes for the instance to be ready for you to connect to it.



Deploying Websites through EC2

Static

1. SSH into EC2 Instance



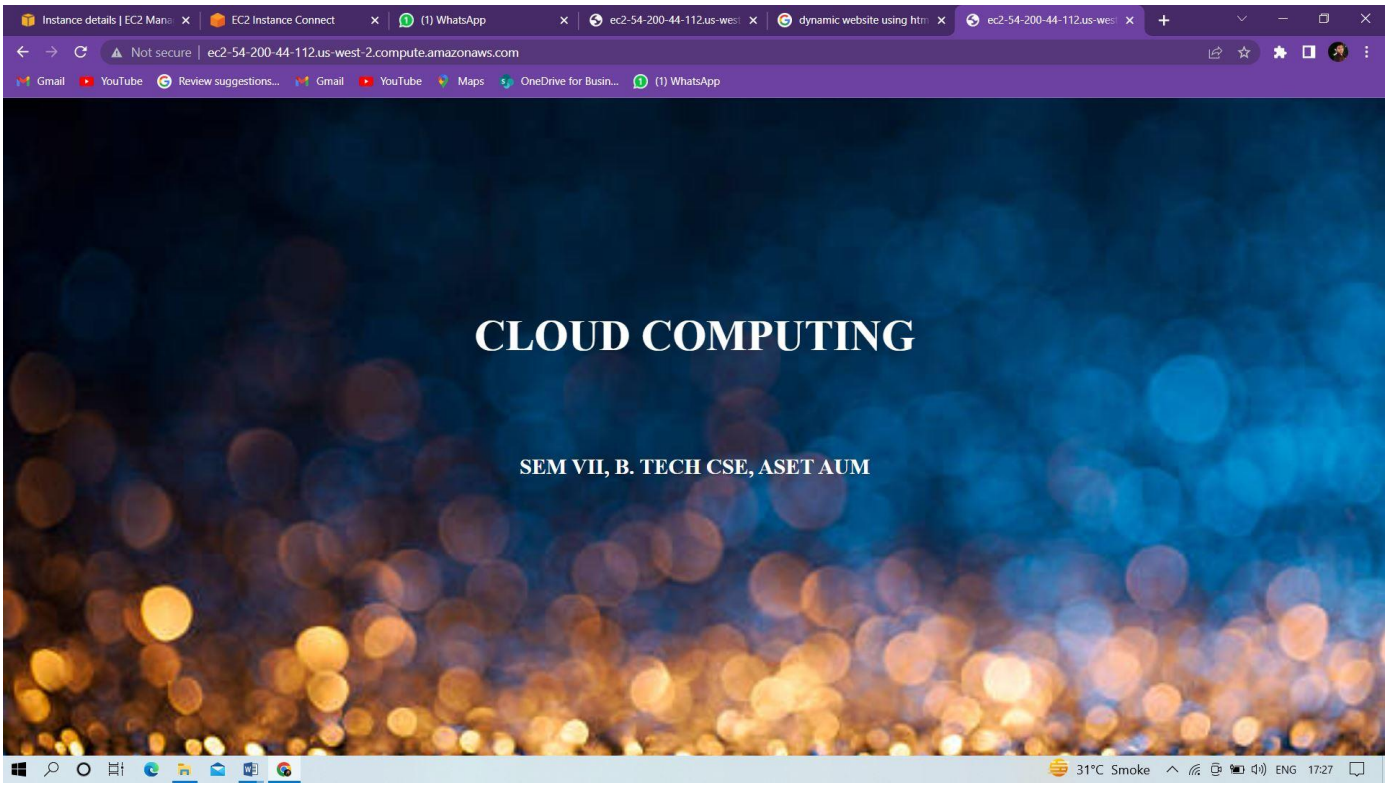
2. Elevate your privileges: **sudo su**
3. Update all the packages on the instance: **yum update -y**
4. Install an Apache webserver: **yum install httpd -y**
5. Navigate to **/var/www/html**

```
Last login: Sat Dec  3 10:50:16 2022 from ec2-18-237-140-163.us-west-2.compute.amazonaws.com

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-3-126 ~]$ sudo su
[root@ip-172-31-3-126 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                 | 3.7 kB  00:00:00
No packages marked for update
[root@ip-172-31-3-126 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package httpd-2.4.54-1.amzn2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-3-126 ec2-user]# pwd
/home/ec2-user
[root@ip-172-31-3-126 ec2-user]# ls
[root@ip-172-31-3-126 ec2-user]# cd /var/www/html
[root@ip-172-31-3-126 html]# ls
index.html
[root@ip-172-31-3-126 html]#
```

6. Copy the index.html file from S3 bucket to EC2 instance:
   s3://my-awsbucketnew/index.html .
7. Start the webserver: service httpd start

```
[root@ip-172-31-3-126 html]# aws s3 cp s3://my-awsbucketnew/index.html .
download: s3://my-awsbucketnew/index.html to ./index.html
[root@ip-172-31-3-126 html]# ls
index.html
[root@ip-172-31-3-126 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-3-126 html]#
```

8. Navigate back to the EC2 dashboard in the AWS console and copy the public DNS (IPv4) of the instance. Paste the address with http:// at the start.



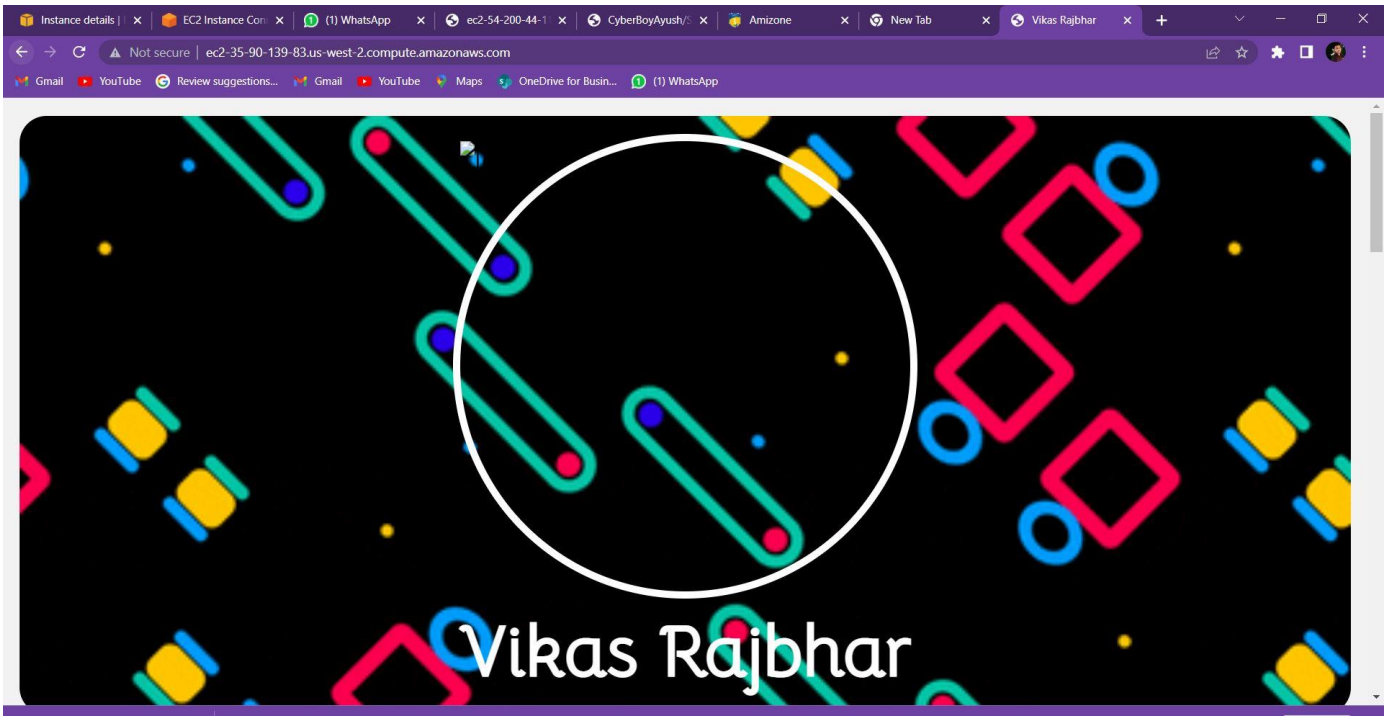
Dynamic

The steps to deploy a dynamic website are like the steps mentioned previously, except for copying the files and folders.

The **--recursive** flag is used in the copying command to recursively copy the files from directories, thus, copying whole directories. Copy all the files and folders into **/var/www/html**

1. aws s3 cp s3://vikas-website/ . --recursive

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-12-139 ~]$ sudo su
[root@ip-172-31-12-139 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
No packages marked for update
[root@ip-172-31-12-139 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package httpd-2.4.54-1.amzn2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-12-139 ec2-user]# ls
[root@ip-172-31-12-139 ec2-user]# cd /var/www/html
[root@ip-172-31-12-139 html]# ls
[root@ip-172-31-12-139 html]# aws s3 cp s3://vikasbucket1278/vikas-website/ . --recursive
download: s3://vikasbucket1278/vikas-website/index.html to ./index.html
download: s3://vikasbucket1278/vikas-website/style.css to ./style.css
[root@ip-172-31-12-139 html]# systemct1 start httpd
bash: systemct1: command not found
[root@ip-172-31-12-139 html]# ls
index.html  style.css
[root@ip-172-31-12-139 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-12-139 html]#
```



Shifting Files

In AWS technical terms.

Copying files from **EC2 to S3** is called **Upload**ing the file

Copying files from **S3 to EC2** is called **Download**ing the files

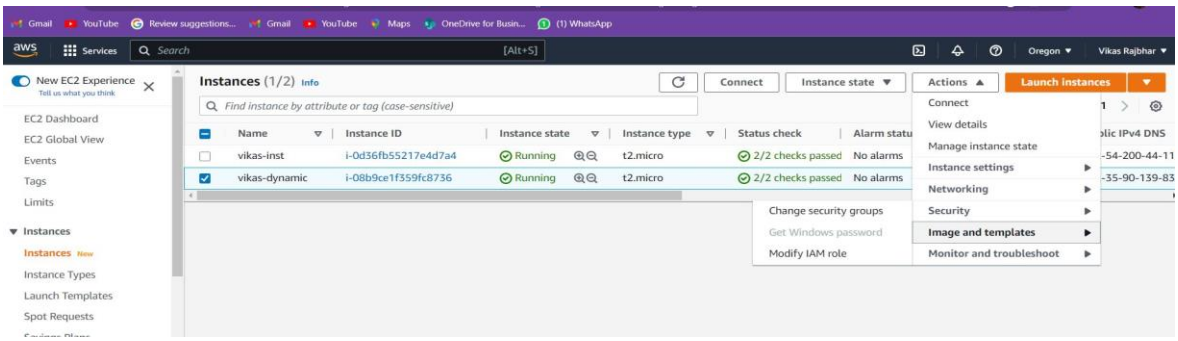
- **S3 to EC2**

  Prerequisites

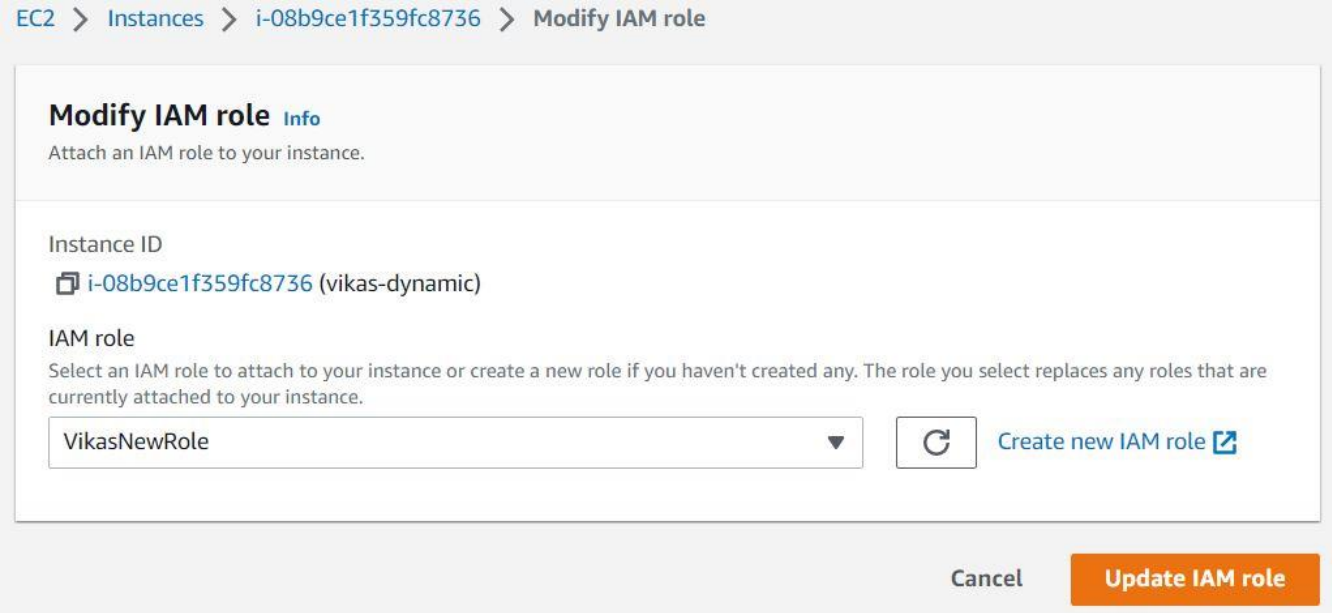
  1. An EC2 instance should exist where we would be performing actions on the S3 bucket
  2. Ensure that AWS CLI is installed on the EC2 instance
  3. An S3 bucket with some files to use for the transfer

  Perform the following steps to add an instance profile to the EC2 instance to grant required permissions to the instance:

  1. Log in to the AWS console
  2. Navigate to the EC2 details page
  3. Select the instance we want to add the instance profile (IAM role) to, click on the Actions drop-down menu, and click on the “Modify IAM role” option



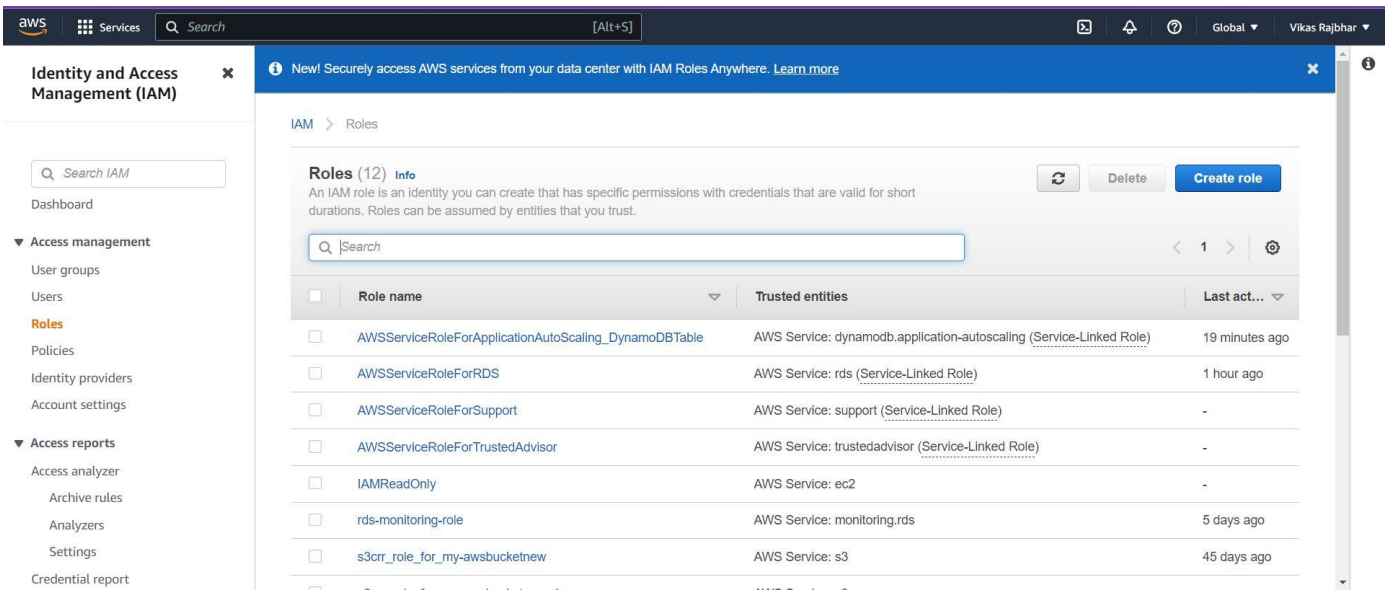
4. Select an existing IAM role from the drop-down menu (or create a new one if needed) and click on the "Save" button.



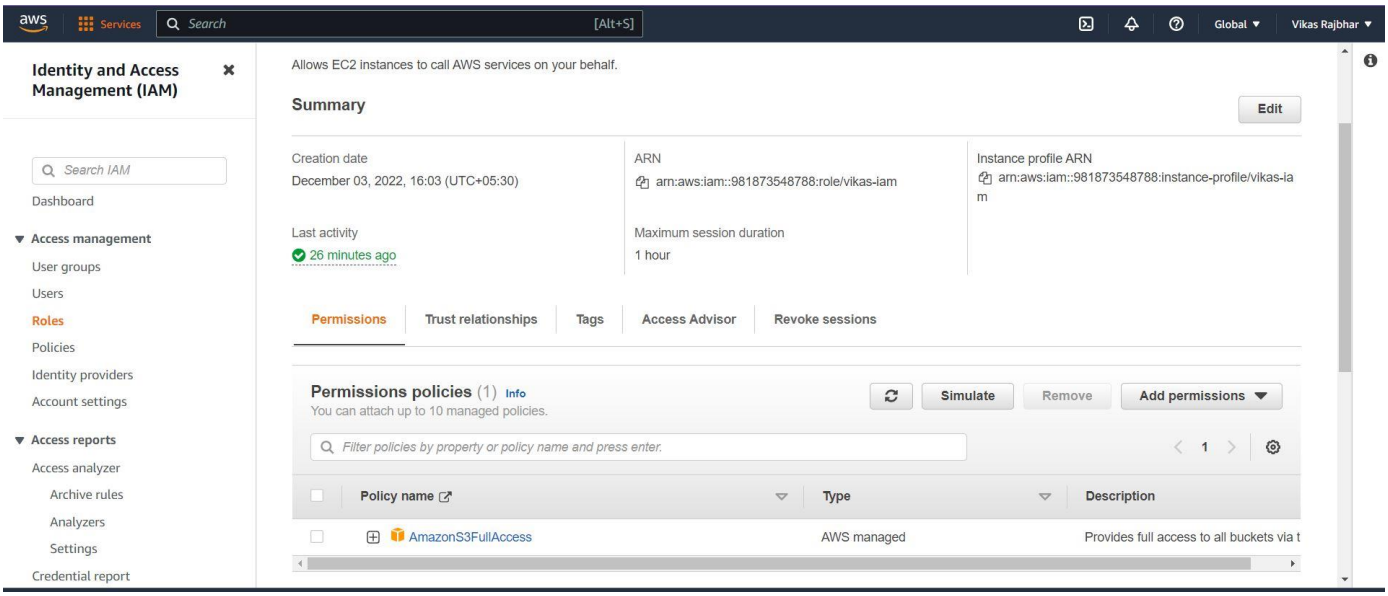
Adding required S3 permissions to the IAM Instance Profile role

Perform the following steps to add required permission to the IAM instance profile role to allow the instance to perform actions on the S3 bucket

1. Navigate to the IAM details page and click on the roles menu and select the role that was attached to the EC2 instance



2. Click on the "Add permissions" button and select the "Attach policies" option
3. Filter and attach the AmazonS3FullAccess policy from the list and click on the Attach policies button



Copying files from the S3 bucket to the EC2 instance

Perform the following steps to copy a file from an S3 bucket to the EC2 instance:

1. SSH into the EC2 instance (using PuTTY)
2. Run `aws s3 cp <S3\_Object\_URI> <Local\_File\_Path>` to copy files from S3 bucket to the EC2 instance

```
[root@ip-172-31-12-139 html]# aws s3 cp s3://vikasbucket1278/vikas-website/index.html .
download: s3://vikasbucket1278/vikas-website/index.html to ./index.html
[root@ip-172-31-12-139 html]# ls
index.html  style.css
[root@ip-172-31-12-139 html]#
```

- **EC2 to S3**

To move files from EC2 to S3, we simply execute the following command with the previously mentioned IAM roles still being placed under the instances.

1. To list the files, present in the bucket: `aws s3 ls s3://vikasbucket1278`
2. To move a file from EC2 to S3:

```
[root@ip-172-31-12-139 html]# aws s3 ls s3://vikasbucket1278
                           PRE vikas-website/
[root@ip-172-31-12-139 html]# ls
index.html  style.css
[root@ip-172-31-12-139 html]#
```