Host a dynamic website on AWS

**What is Amazon S3?**

* Amazon S3 (Simple Storage Service) is a service offered by AWS for object storage through a web service interface.
* It can be used to store or retrieve any amount of data such as documents, images, videos, etc.
* S3 bucket is a resource in Amazon S3. It is a container where files and folders can be uploaded.

**What is Amazon EC2?**

* Amazon EC2 (Elastic Compute Cloud) is a service offered by AWS. It is considered as a virtual server.

**What is IAM Role?**

* IAM (Identity and access management) Role is used to give permission to service to do something on another service.

**What is LAMP web server?**

* You can use this server to host a static website or deploy a dynamic PHP application that reads and writes information to a database. You can know more from here.

**Table of Contents:**

1. Create S3 Bucket

2.Upload web files to S3 bucket

3.Create IAM Role

4.Create an EC2 instance

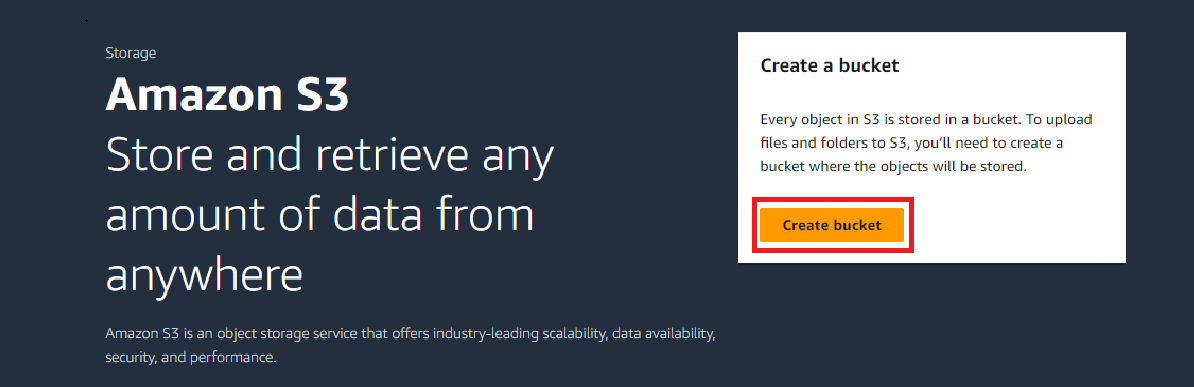
5.Connect instance with sshclient

6.Install a LMP web server on Amazon Linux2

Now, let’s get into it!

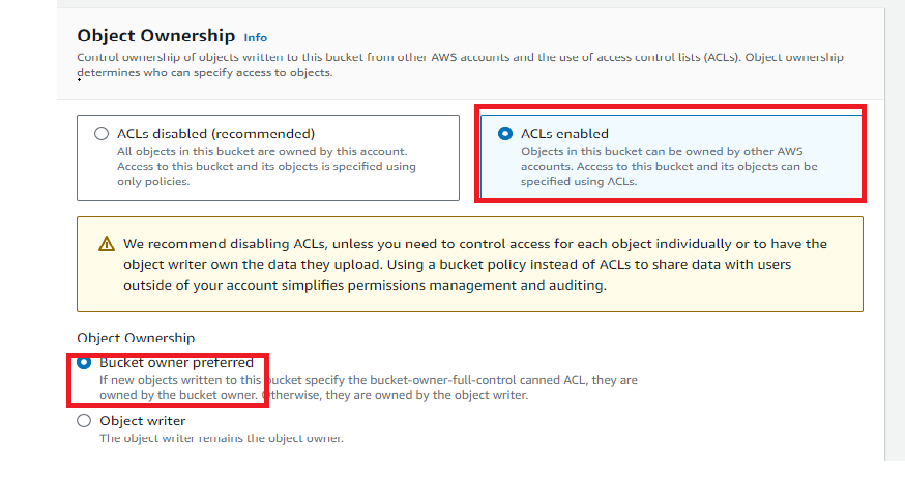
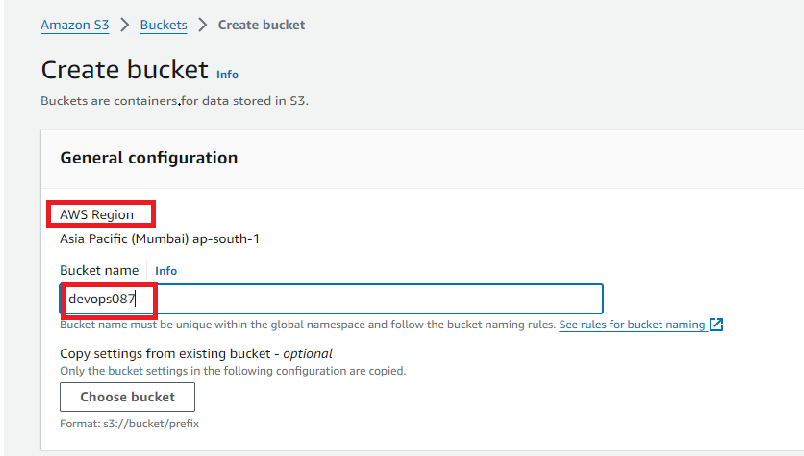
**Step 1: Create S3 Bucket**

You will need to create an S3 bucket to put your website’s files and folders.

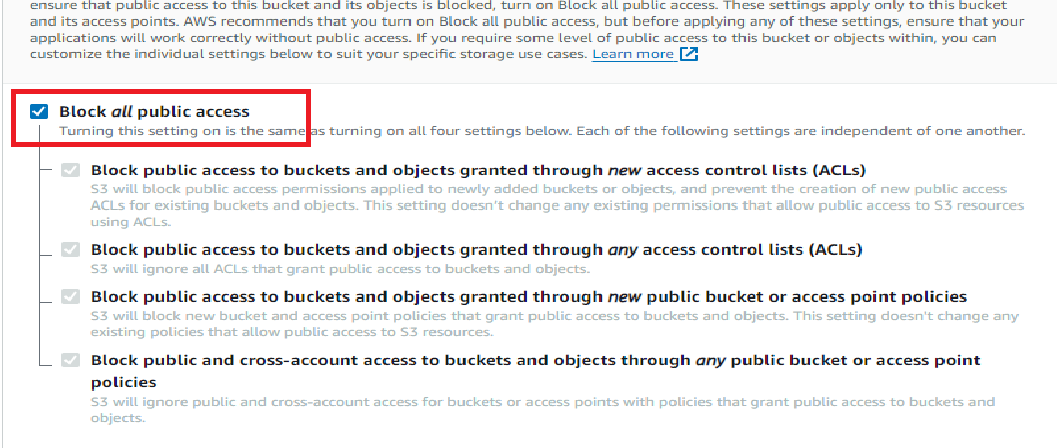
To do this, login into your AWS management console and click on Services on the top navbar. From the Services drop-down, select S3 from the Storage section. This should display the S3 dashboard.

From the S3 dashboard, click on Create bucket. Give the bucket a unique name, the name you choose must be globally unique.

Next, choose your preferred AWS Region from the drop-down.

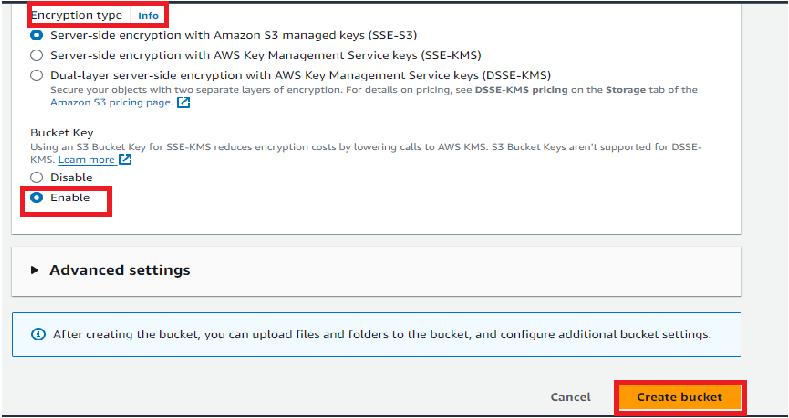


Under Block Public Access settings for this bucket section, check the Block all public access checkbox. This is done to make the bucket not accessible to the public.

.

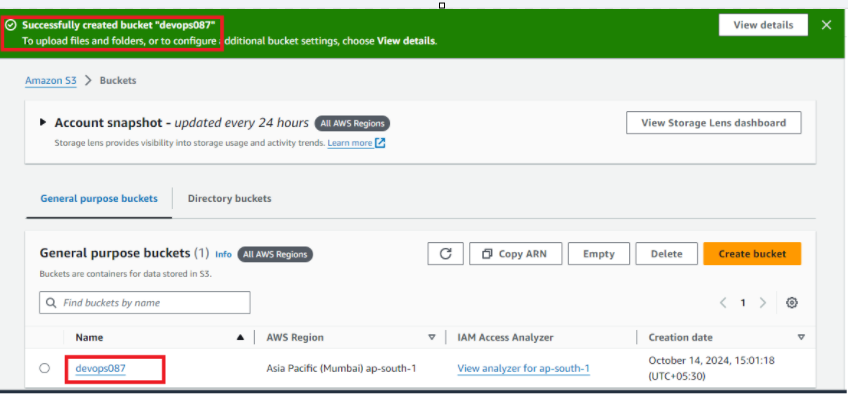
Click on Disable for Bucket Versioning

Under Default encryption section, click on Enable for Server-side encryption. Then check Amazon S3 Key (SSE-S3).



Then click on Create bucket.

The below image shows the successfully created bucket.

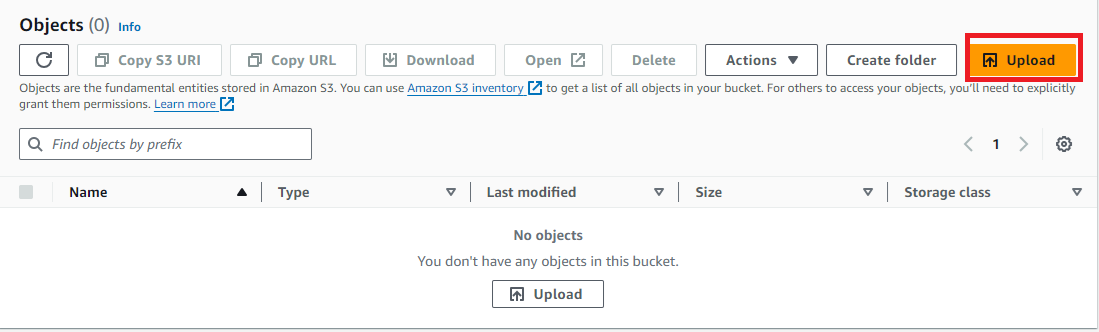


**Step 2: Upload web files to S3 bucket**

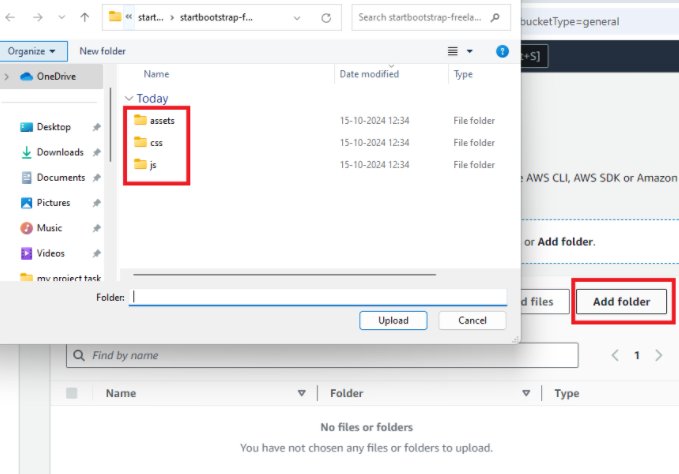
After creating the bucket, you need to upload your website’s files and folders into it.

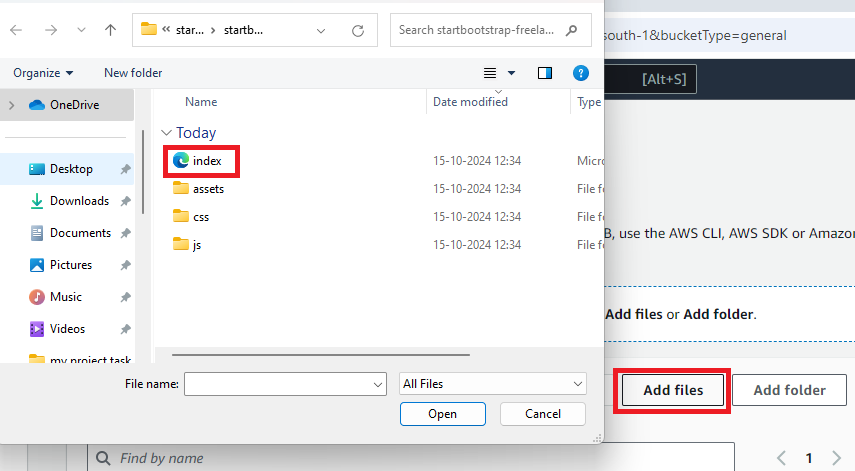
From the S3 dashboard, click on the name of the bucket you just created.

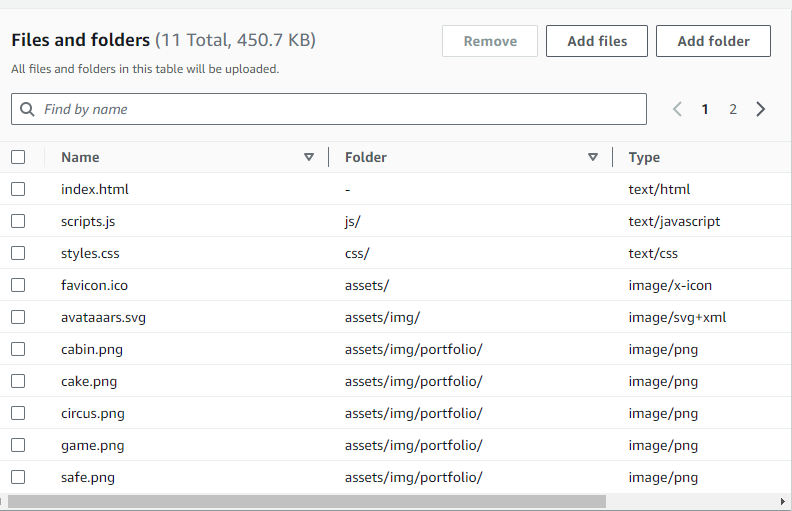
On the Objects tab, you can see that the bucket is currently empty, click on the Upload button.



This should take you to the Upload page. Drag and Drop the website files that was downloaded from this.







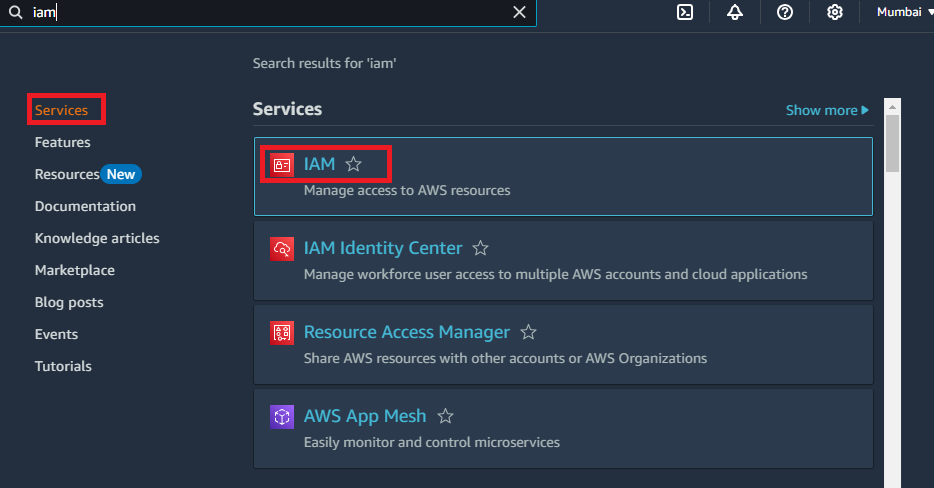
After the necessary files and folders have been added, scroll down and click on Upload.

The uploading should be done in a few minutes depending on your network and content size. Also, please do not close the tab while the upload process is going on.

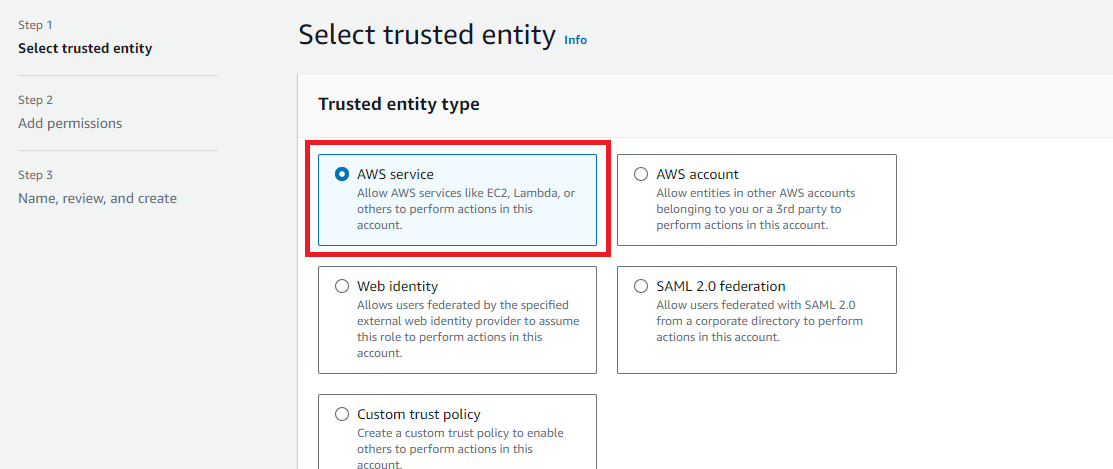
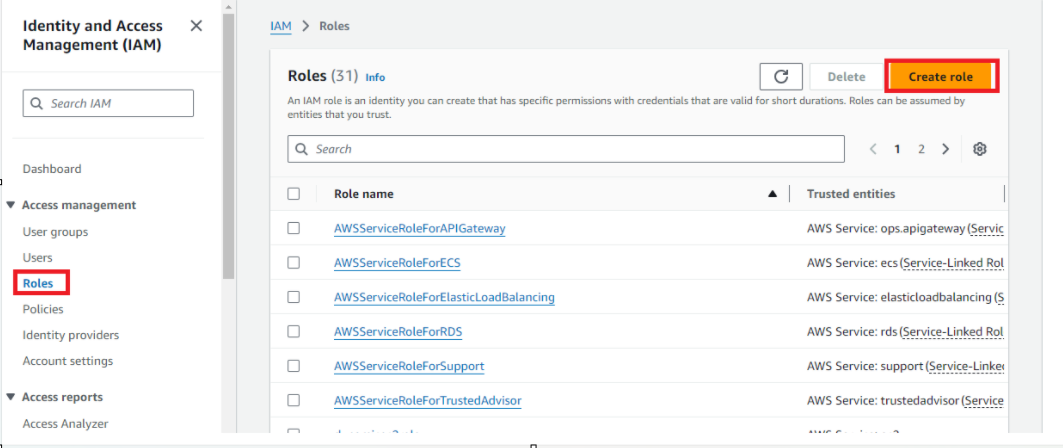
**Step 3: Create IAM Role**

Now, EC2 want to pull code from S3.So, you want to create IAM Role to give EC2 permission to access S3.

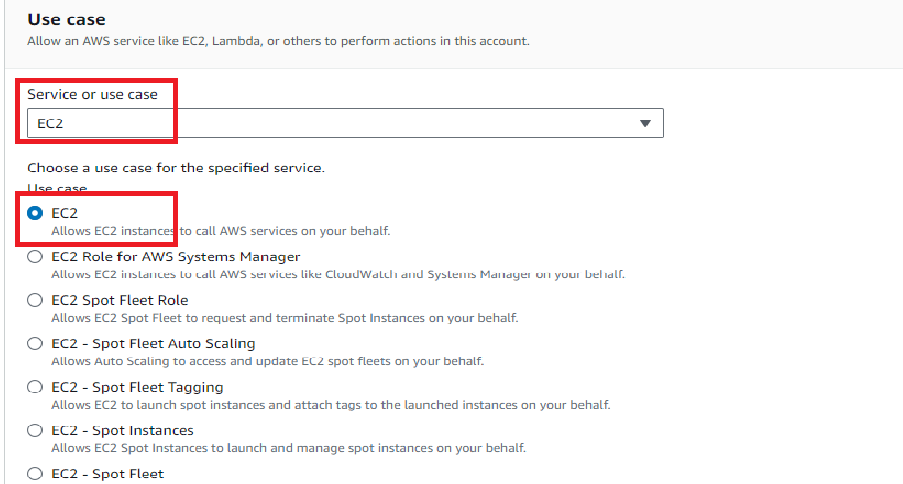
To do this, from the Services drop-down, select IAM from the Security, Identity& Compliance section.

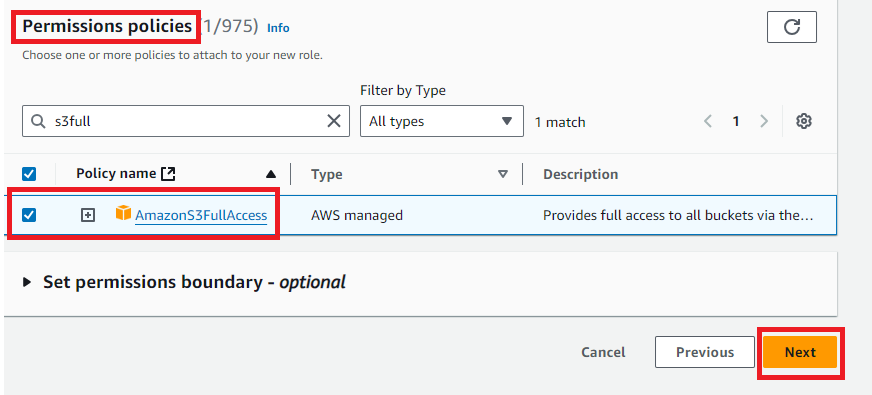


From the IAM dashboard, click on Roles. Then Click on Create role.

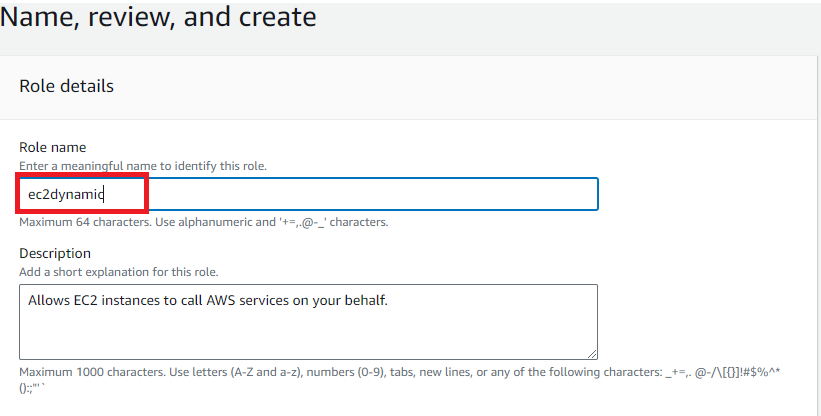


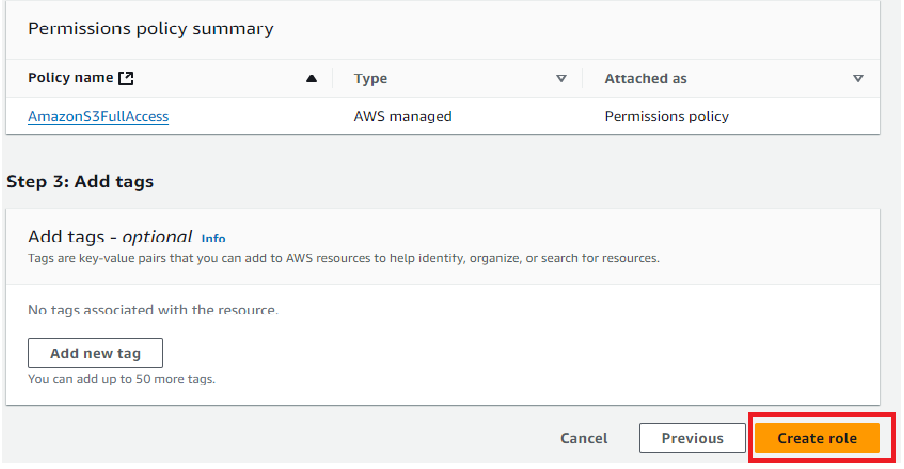
Choose EC2 and click Next: Permissions.





Give the role name and description. Then click on Create role.



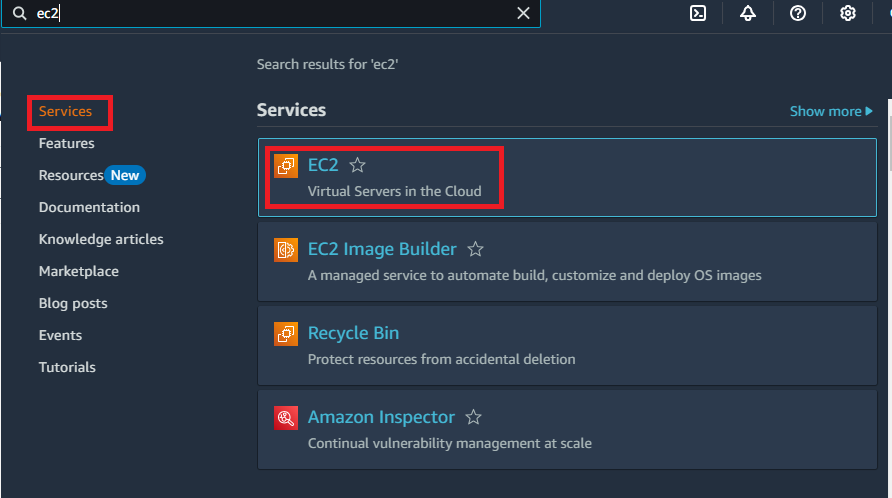


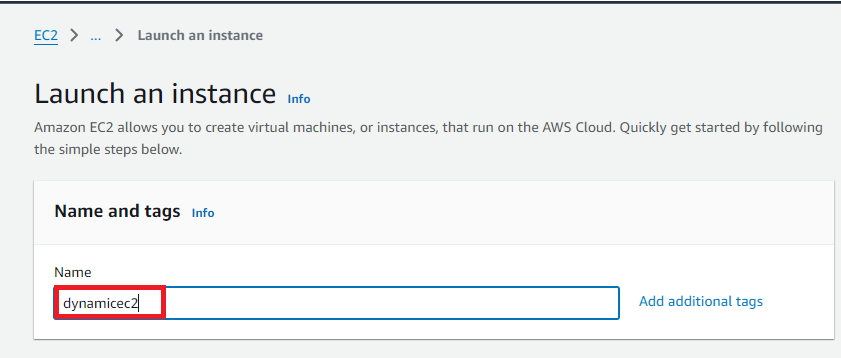
Now, the role has been created successfully.

**Step 4: Create an EC2 instance**

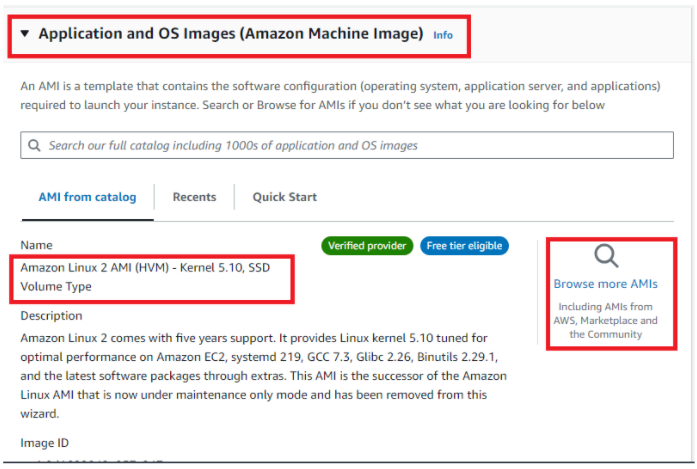
You will need to create an EC2 instance to install apache ( /var/www/html) and copy the content of S3 to html directory.

To do this, from the Services drop-down, select EC2 from the Compute section. This should display the EC2 dashboard. From the EC2 dashboard, click on Launch instance.

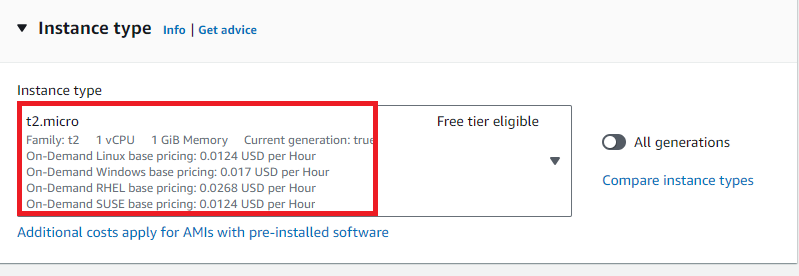




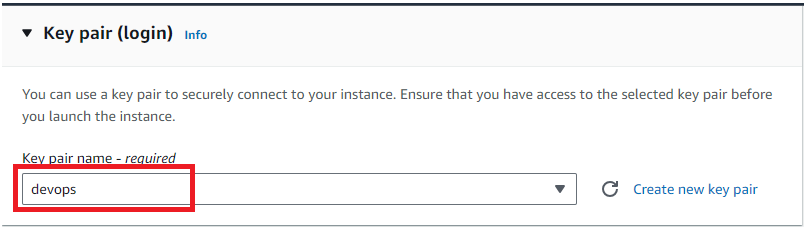
For AMI, choose Quick Start and click on Select for Amazon Linux 2(Free tier eligible).



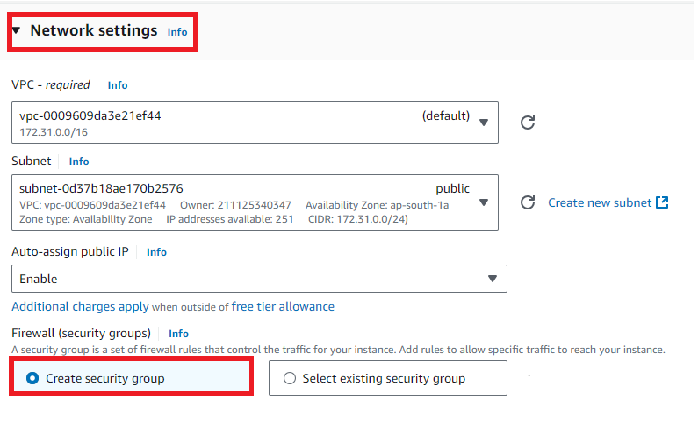
For an instance type, choose t2. micro (Free tier eligible). And click on Next: Configure Instance Details.



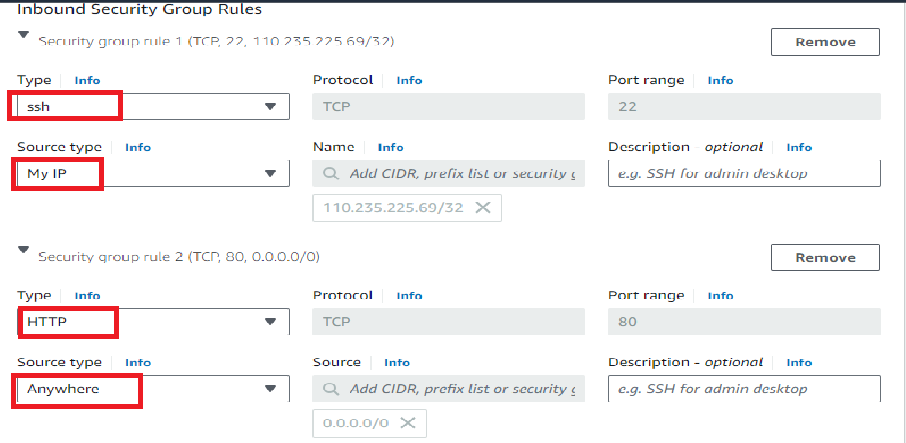
Select Create a new key pair and RSA for type. Give it name devops and click on Download Key Pair. Note: You should download the key to can ssh on EC2. Click on Launch Instances.

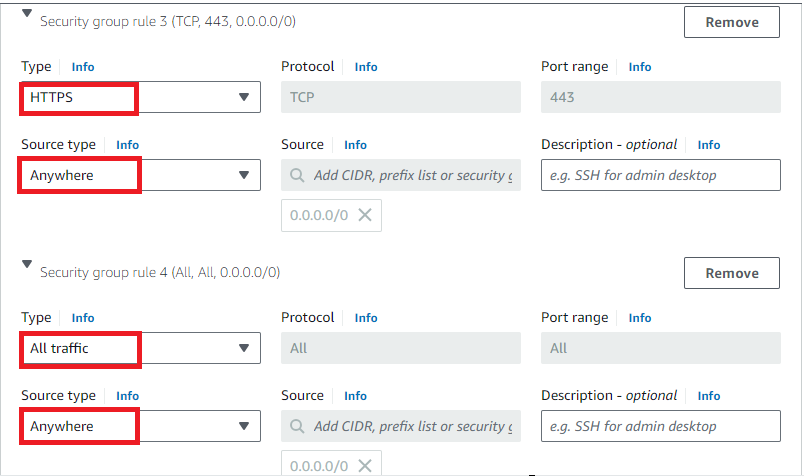


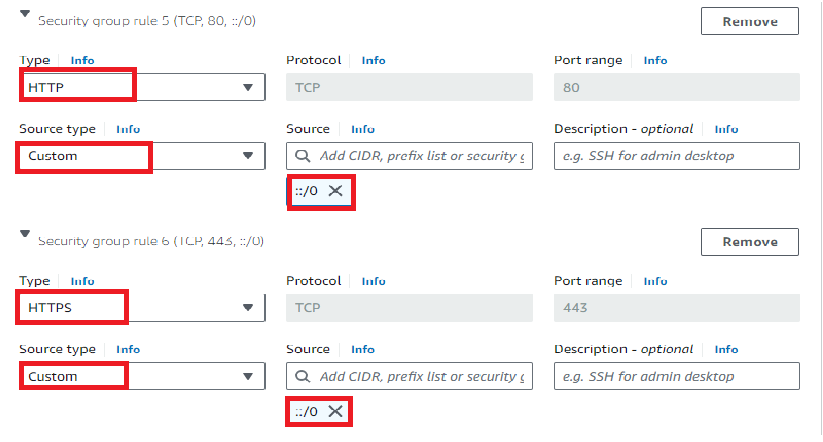
Determine 1 for Number of instances, default vpc for Network and Default in ap-south-1a for Subnet.



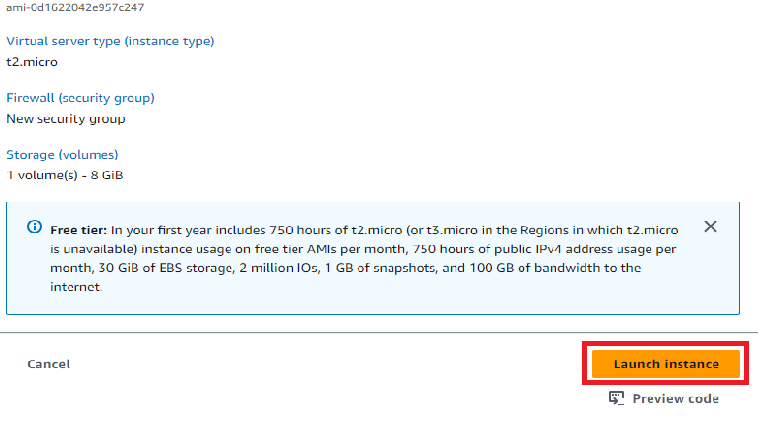
For SSH rule select My IP for Source. Click on Add Rule and select HTTP for Type and Anywhere for Source. Last rule selects HTTPS for Type and Anywhere for Source.

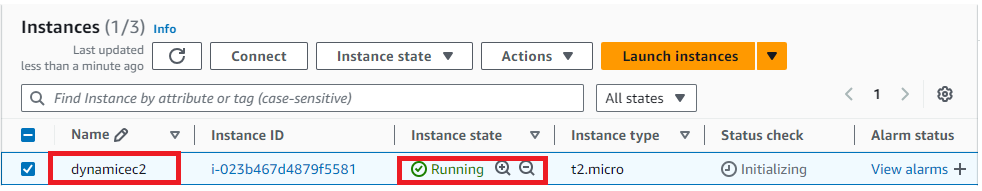




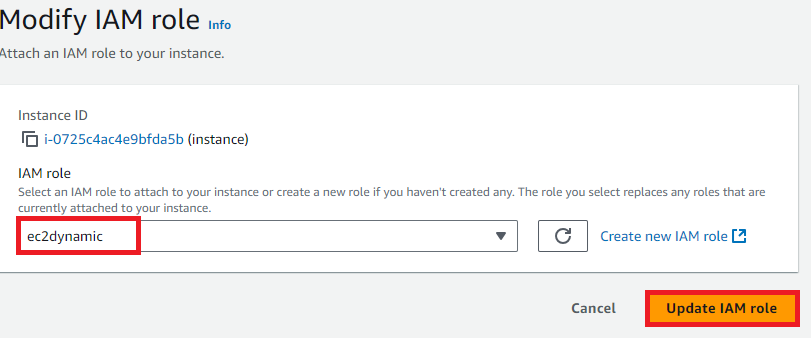


Click on the launch instance.

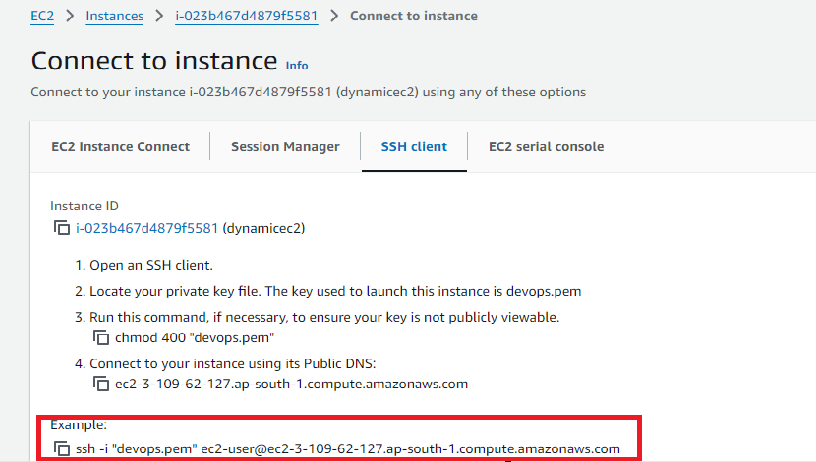




The above image shows successfully created instance.

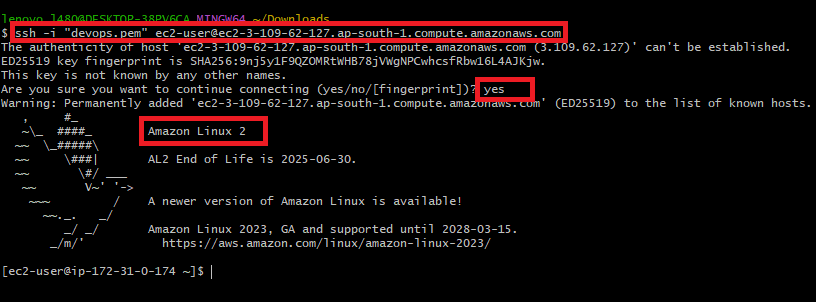


**Step 5: connect instance with sshclient**



Go to downloads and click on devops.pem and open git bash here copy the ssh url in instance and paste it on git bash.

Here the below image shows how we connect to git bash.



**Step 6: Install a LAMP web server on Amazon Linux 2**

* The following procedures help you install an Apache web server with PHP and MariaDB.
* To ensure that all of your software packages are up to date, perform a quick software update on your instance.

**sudo yum update -y**

Install the lamp-mariadb10.2-php7.2 and php7.2 Amazon Linux Extras repositories to get the latest versions of the LAMP MariaDB and PHP packages for Amazon Linux 2.

**sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2**

Now, you can install the Apache web server, MariaDB, and PHP software packages.

**sudo yum install -y httpd mariadb-server**

Start the Apache web server.

**sudo systemctl start httpd**

Use the systemctl command to configure the Apache web server to start at each system boot.

**sudo systemctl enable httpd**

You can verify that httpd is on by running

**sudo systemctl is-enabled httpd**

Now, you want to copy content of website from S3 to directory /var/www/html in EC2 . Make sure you copy your S3 bucket name.

**sudo aws s3 cp s3://devops087 --region ap-south-1 /var/www/html/ --recursive**

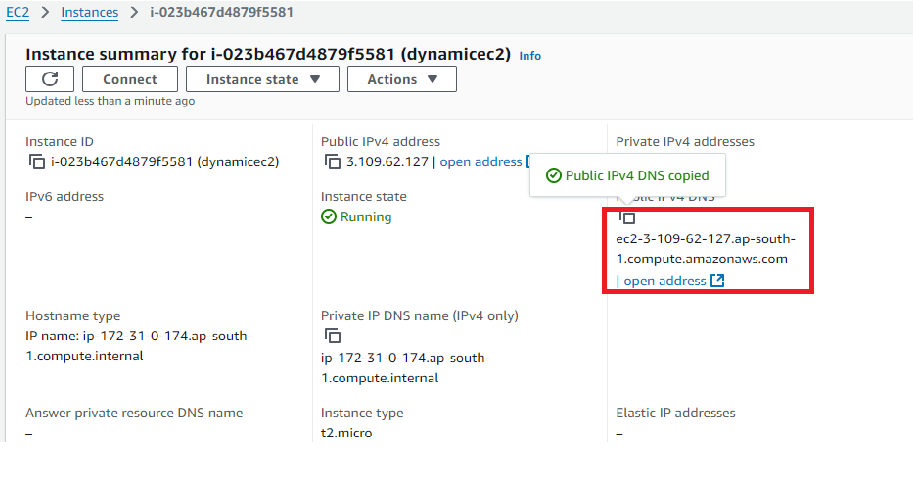
To verify that content is copied to /var/www/html.

**cd /var/www/html**

**ls**



Copy public IPv4 DNS and paste it in a new tab.



we have deployed a dynamic website on EC2 successfully.

