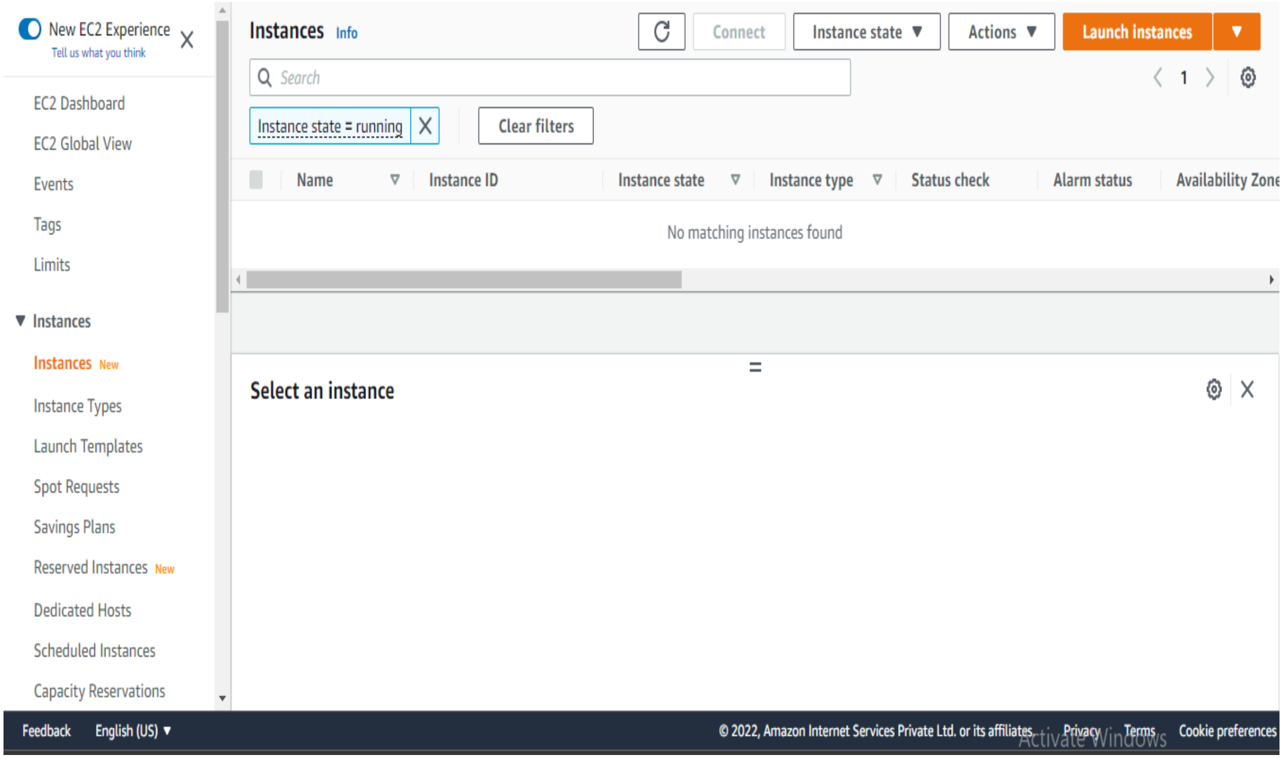
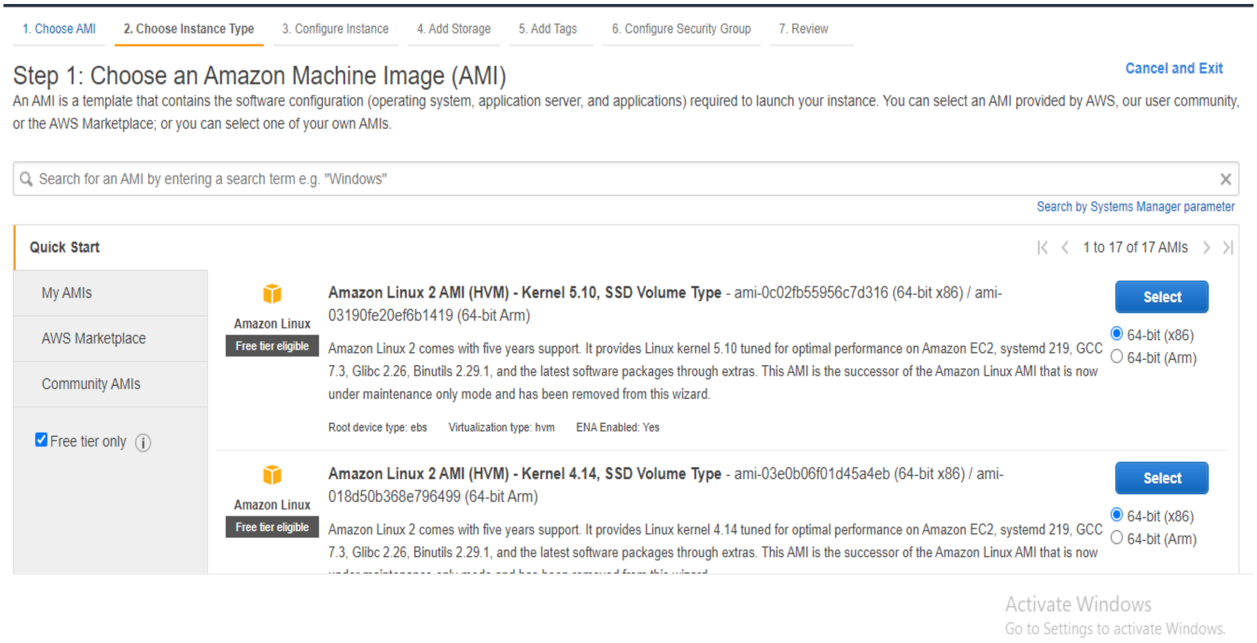
Static website on ec2(AWS)

Launch an EC2 instance

1. Sign in to AWS console and open the ec2 dashboard.



2. Choose the option to Launch Instance.
   2.1 Choose the free tier option to avoid chargers for machine.
   2.2 Under free tier you can use 750hrs an ec2 instance.
   2.3 Now choose the Amazon Linux 2 AMI instance and do further
   do further configuration in the next steps.



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

☑ Free tier only

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type - ami-0c02fb55956c7d316 (64-bit x86) / ami-03190fe20ef6b1419 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

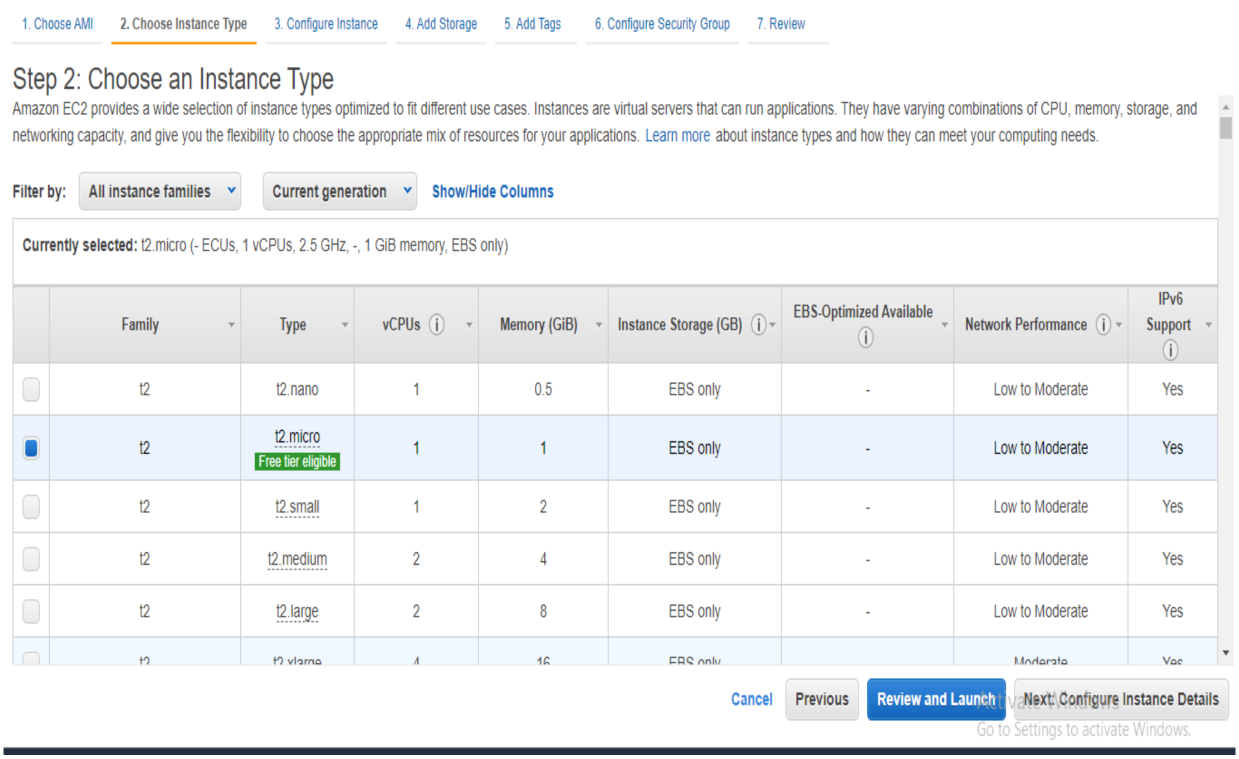
● 64-bit (x86)
○ 64-bit (Arm)

Amazon Linux 2 AMI (HVM) - Kernel 4.14, SSD Volume Type - ami-03e0b06f01d45a4eb (64-bit x86) / ami-018d50b368e796499 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

● 64-bit (x86)
○ 64-bit (Arm)

3. Choose the **t2.micro** instance type, as shown following, and then choose **Next: Configure Instance Details**.



Step 2: Choose an Instance Type

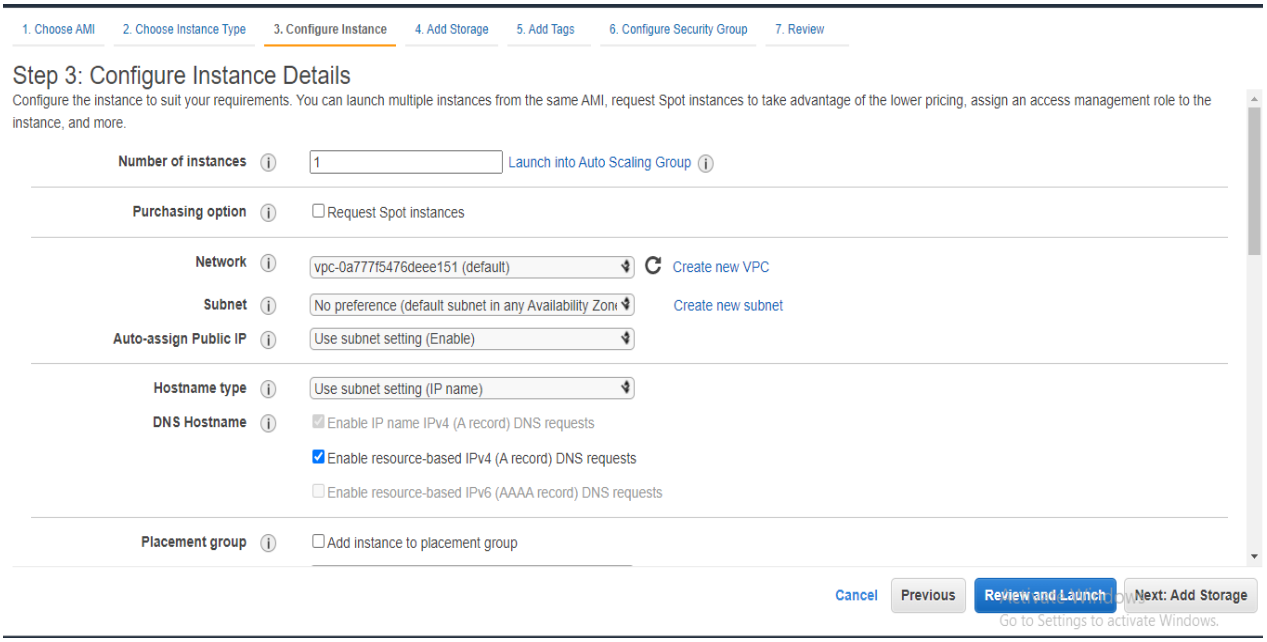
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance families | Current generation | Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
□	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
■	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
□	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
□	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
□	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
□	t2	t2.xlarge	4	16	EBS only		Moderate	Yes

4 . On the Configure Instance Details page as their defaults:



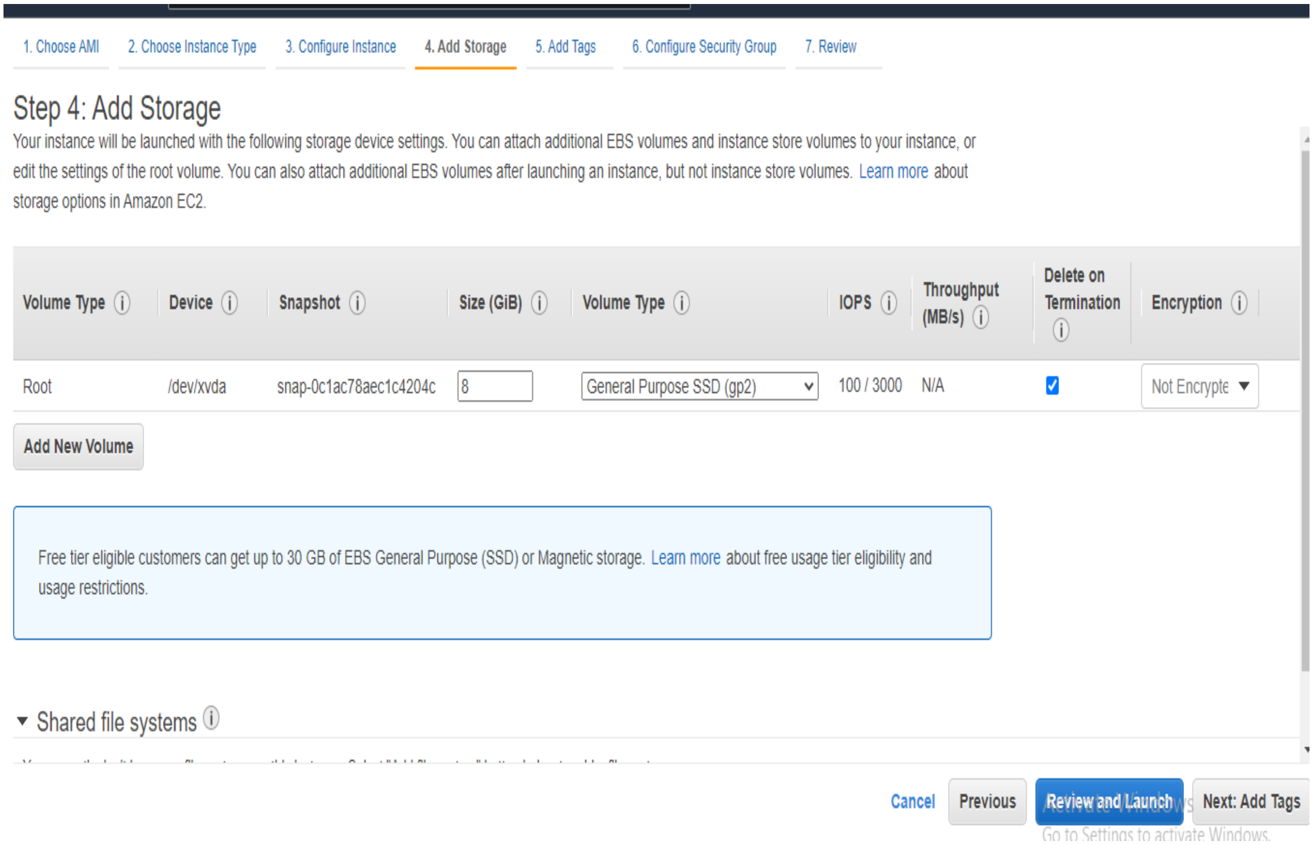
Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1 Launch into Auto Scaling Group
Purchasing option	□ Request Spot instances
Network	vpc-0a777f5476deee151 (default) Create new VPC
Subnet	No preference (default subnet in any Availability Zone) Create new subnet
Auto-assign Public IP	Use subnet setting (Enable)
Hostname type	Use subnet setting (IP name)
DNS Hostname	☑ Enable IP name IPv4 (A record) DNS requests ☑ Enable resource-based IPv4 (A record) DNS requests □ Enable resource-based IPv6 (AAAA record) DNS requests
Placement group	□ Add instance to placement group

5. Choose Next: Add Storage

   5.1 In the Add storage page you can extra ebs volume if you want by add volume Option.



Step 4: Add Storage

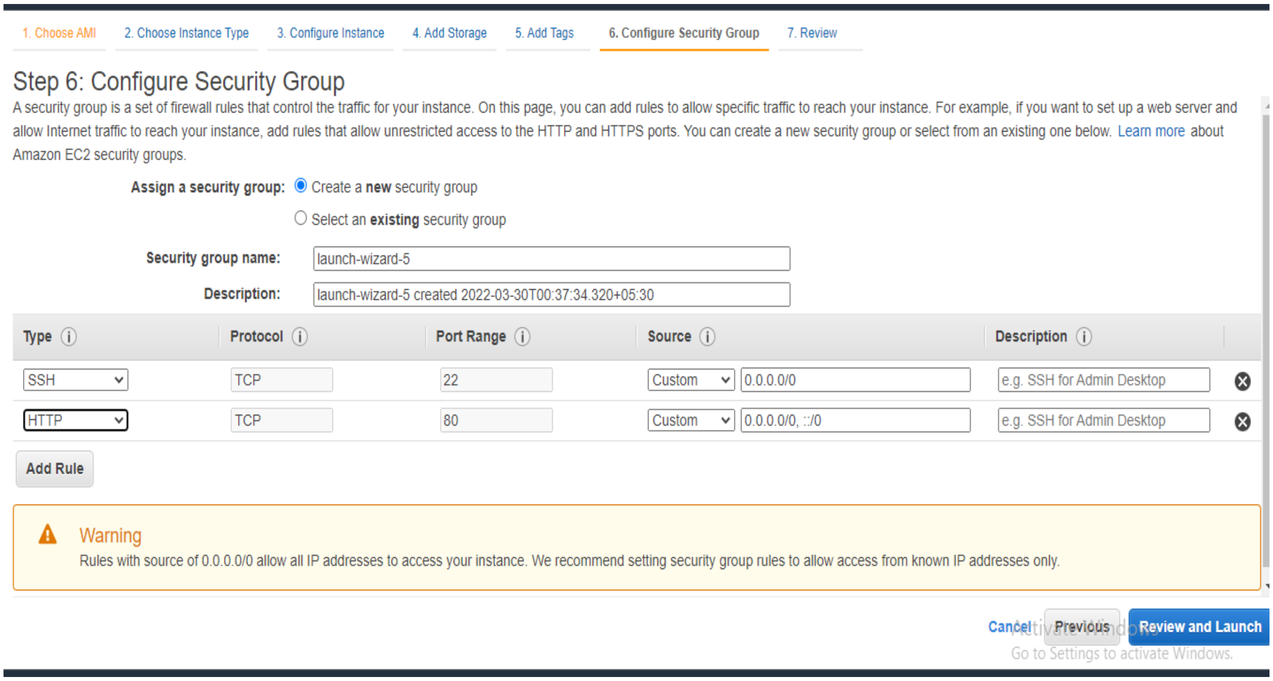
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-0c1ac78aec1c4204c	8	General Purpose SSD (gp2)	100 / 3000	N/A	☑	Not Encrypte

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

▾ Shared file systems

6. On the Configure Security Group page, Make sure that the security group that you choose includes inbound rules for Secure Shell (SSH) and HTTP access.



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

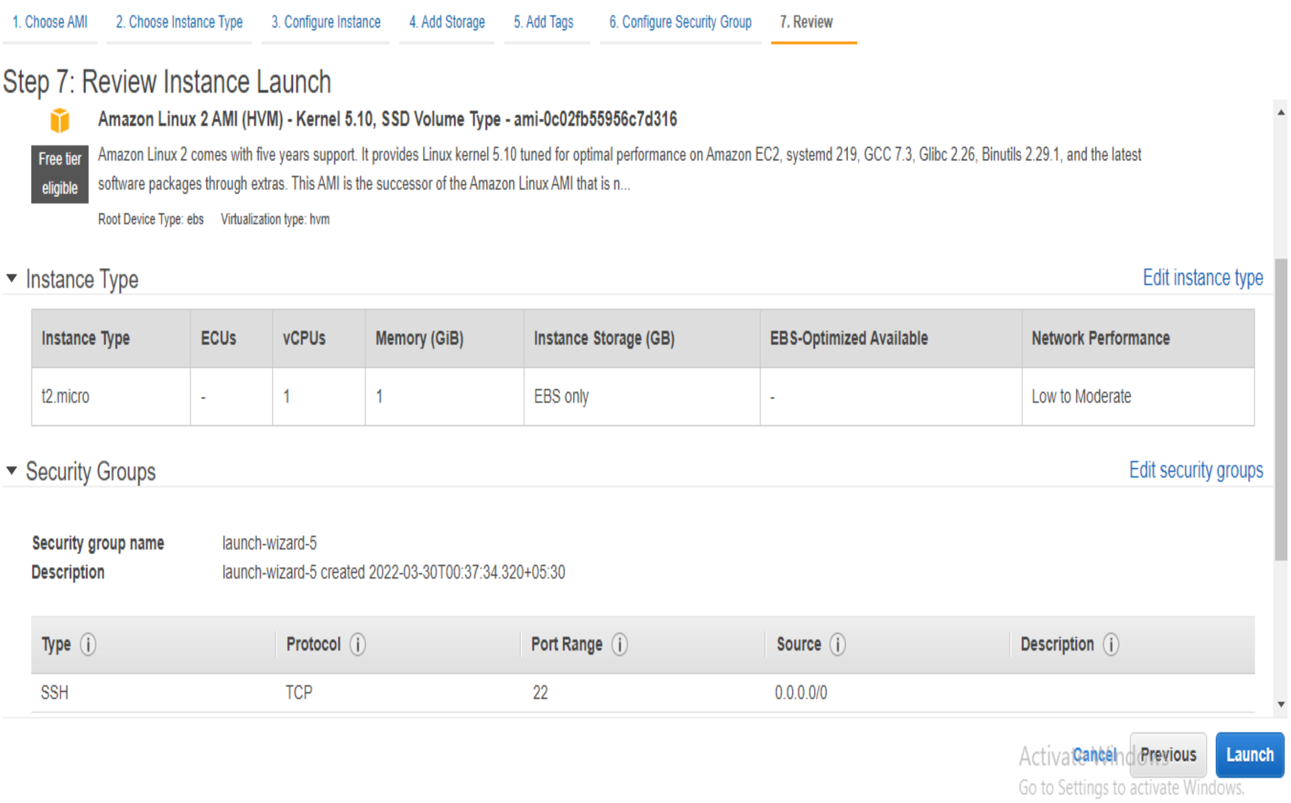
Security group name: launch-wizard-5

Description: launch-wizard-5 created 2022-03-30T00:37:34.320+05:30

Type	Protocol	Port Range	Source		Description
SSH	TCP	22	Custom	0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom	0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

7. Choose Review and Launch.On the Review Instance Launch page, shown following, verify your settings and then choose Launch.



Step 7: Review Instance Launch

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type - ami-0c02fb55956c7d316

Free tier eligible

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is n...

Root Device Type: ebs Virtualization type: hvm

Instance Type

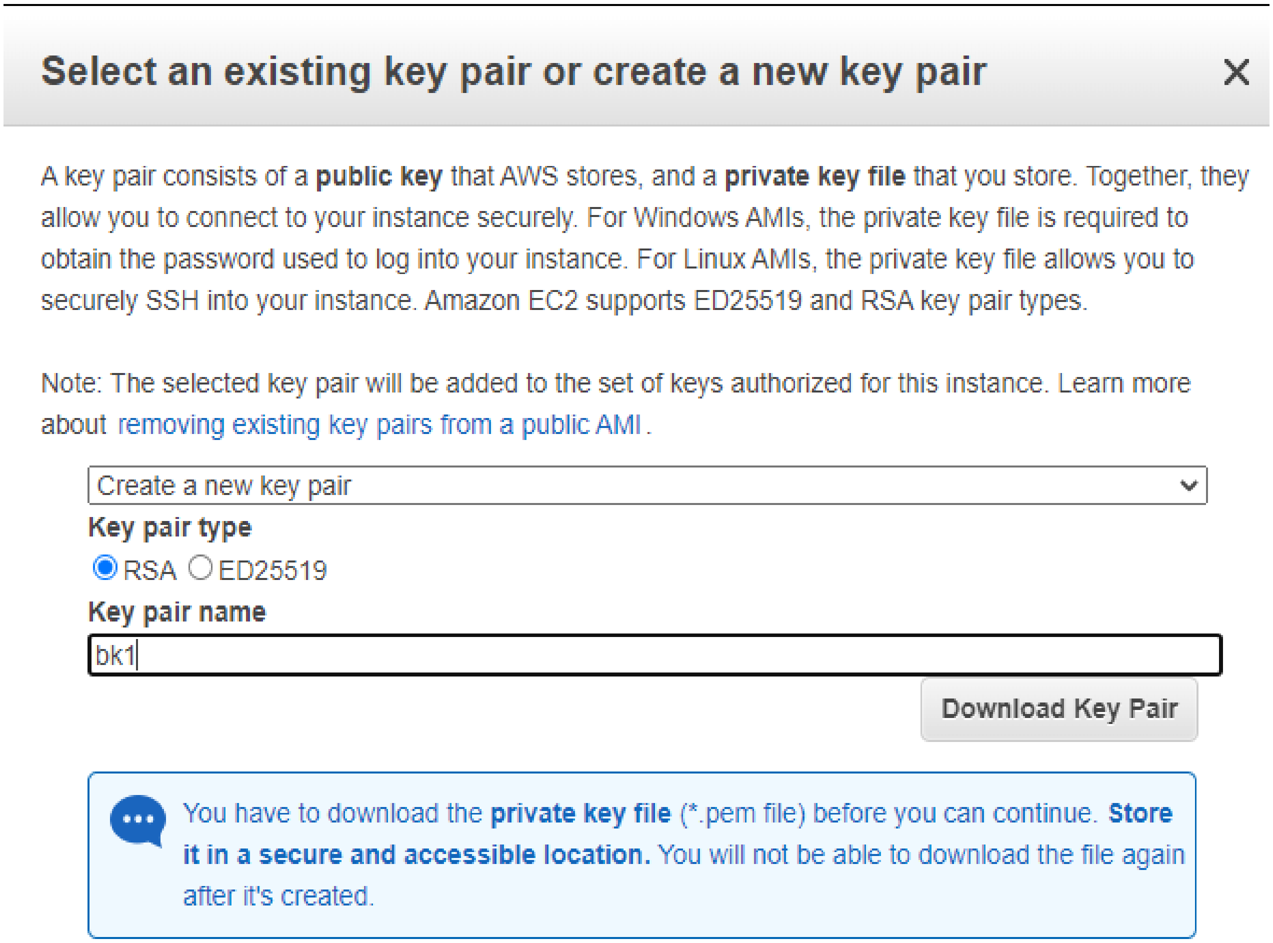
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name launch-wizard-5
Description launch-wizard-5 created 2022-03-30T00:37:34.320+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	

8. On the Select an existing key pair or create a new key pair page, shown following, choose Create a new key pair and set Key pair name. Choose Download Key Pair, and then save the key pair file on your local machine.



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI .

Create a new key pair

Key pair type

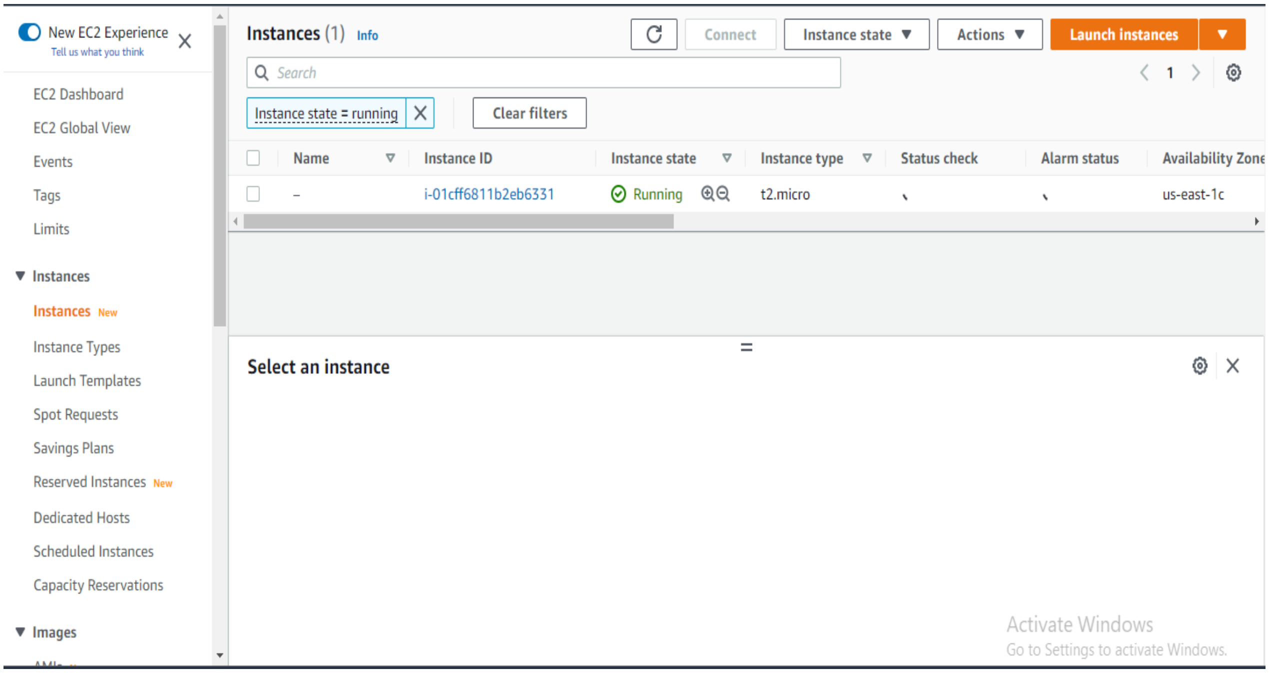
● RSA ○ ED25519

Key pair name

bk1

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

9. To launch your EC2 instance, choose Launch Instances.And you can check the status of in ec2 dashboard.



Instances (1) Info

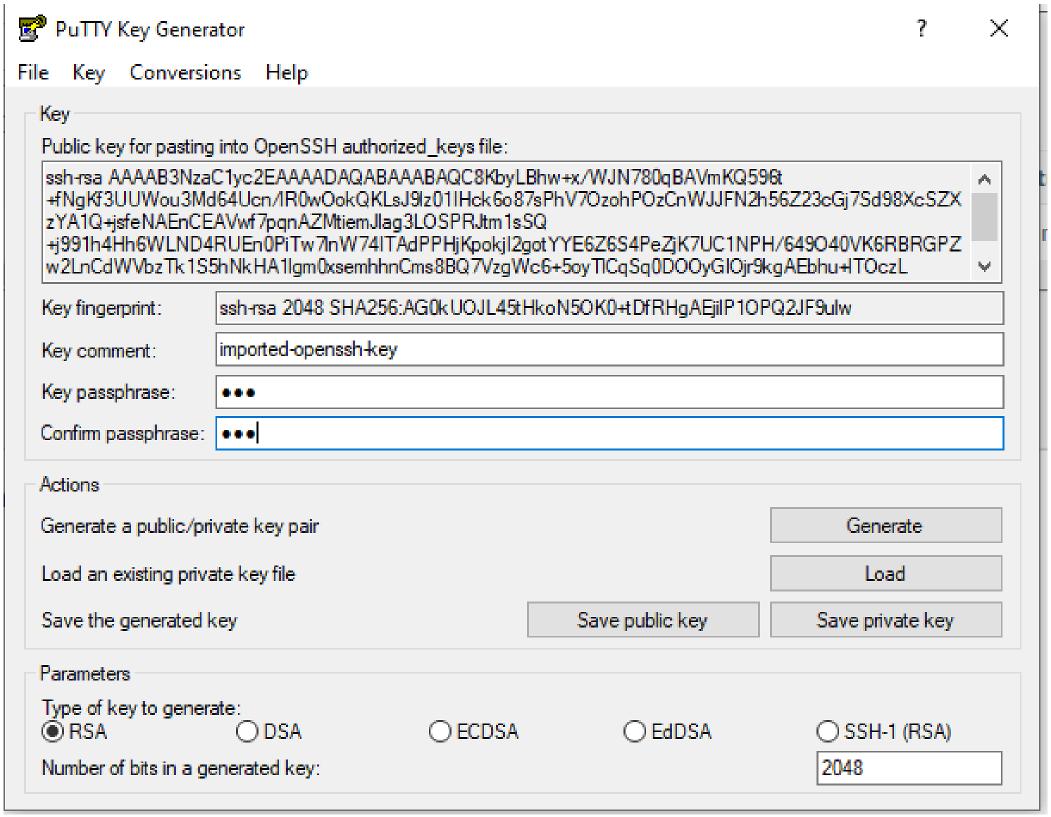
Instance state = running

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
–	i-01cff6811b2eb6331	✓ Running	t2.micro			us-east-1c

Select an instance

To connect to your EC2 instance with the help of putty

1 . Open puttygen and load the key you downloaded during the launchin of ec2 instance and generate private key and save private key to local machine.



Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC8KbyLBhw+x/WJN780qBAVmKQ596t
+fNgKf3UUWou3Md64Ucn/1R0wOokQKLsJ9Iz01lHck6o87sPhV7OzohPOzCnWJJFN2h56Z23cGj7Sd98XcSZX
zYA1Q+jsfeNAEnCEAVwf7pqnAZMtiemJlag3LOSPRJtm1sSQ
+j991h4Hh6WLND4RUEn0PiTw7lnW74lTAdPPHjKpokjl2gotYYE6Z6S4PeZjK7UC1NPH/649O40VK6RBRGPZ
w2LnCdWVbzTk1S5hNkHA1lgm0xsemhhnCms8BQ7VzgWc6+5oyTlCqSq0DOOyGIOjr9kgAEbhu+lTOczL
```

Key fingerprint: ssh-rsa 2048 SHA256:AG0kUOJL45tHkoN5OK0+tDfRHgAEjilP1OPQ2JF9ulw

Key comment: imported-openssh-key

Key passphrase: •••

Confirm passphrase: •••

Actions

Generate a public/private key pair — Generate

Load an existing private key file — Load

Save the generated key — Save public key — Save private key

Parameters

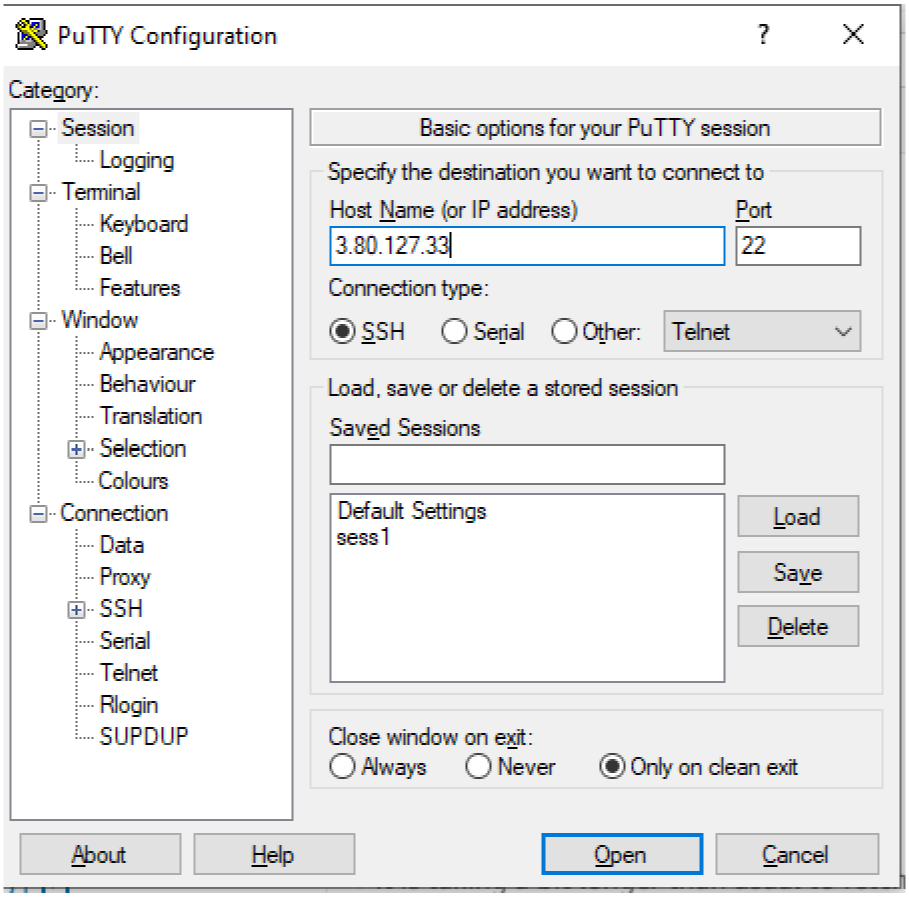
Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

2. Open putty application and paste the public ipv4 address of the ec2 instance in the putty and then go to SSH->Auth on the right side and upload the ppk file which is downloaded in the previos step and then click on the open button.

2.1 login as :- ec3-user

2.2 And write the same passphrase you created in step1(puttyGen) and hit enter.



Specify the destination you want to connect to

Host Name (or IP address): 3.80.127.33

Port: 22

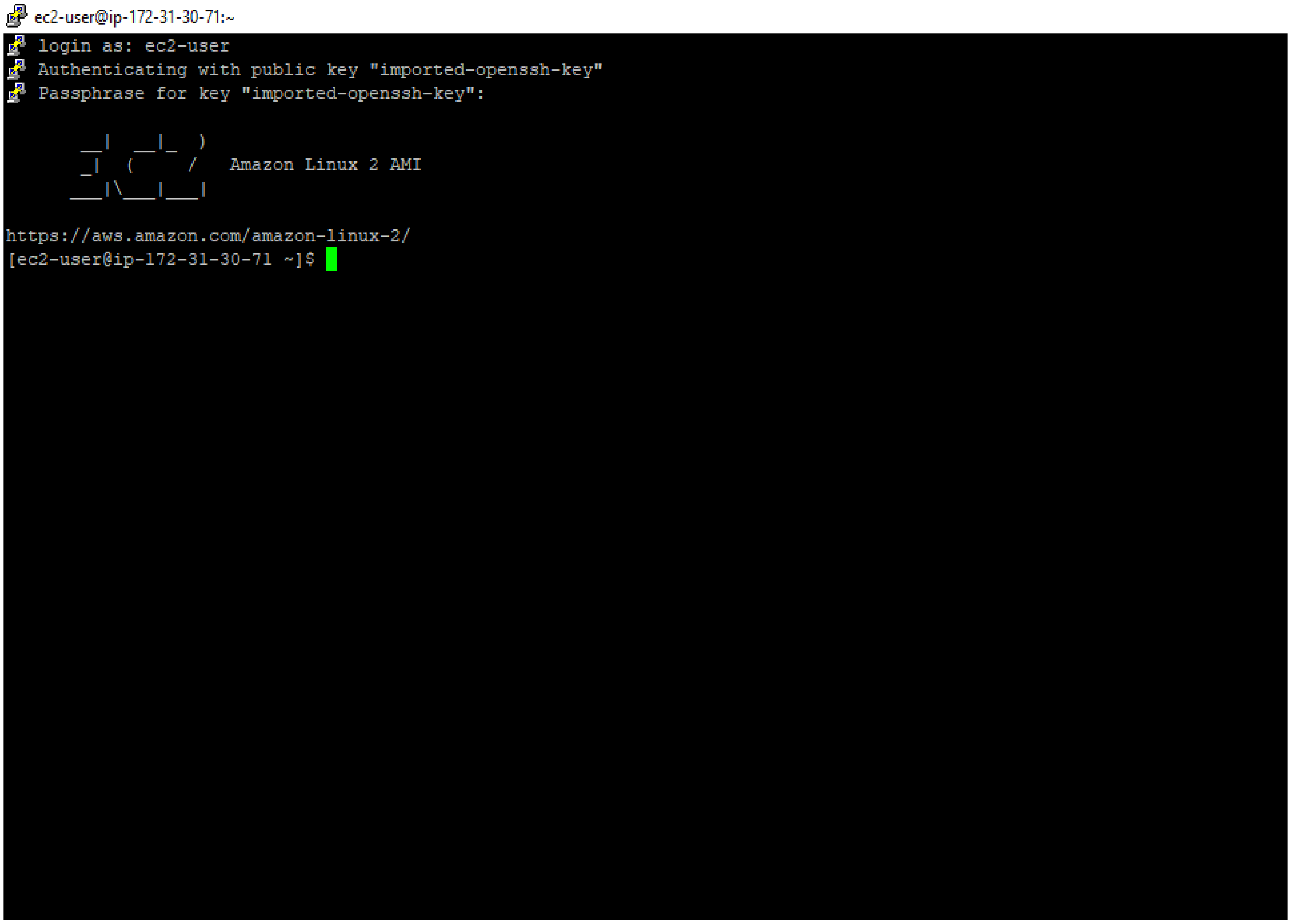
Connection type: ● SSH ○ Serial ○ Other: Telnet

Load, save or delete a stored session

Saved Sessions:

Default Settings
sess1

Close window on exit: ○ Always ○ Never ● Only on clean exit



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Passphrase for key "imported-openssh-key":

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

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-30-71 ~]$
```

3. To run the static website on ec2 instance follow following commands

    3.1 **sudo su**(To go into the the root directory of your linux instance.)

    3.2 **yum update -y(**to update all the resources in the instances.)

    3.3 **yum install httpd -y(**to install the httpd service on the machine)

    3.4 **pwd**(prints the current working directory path)

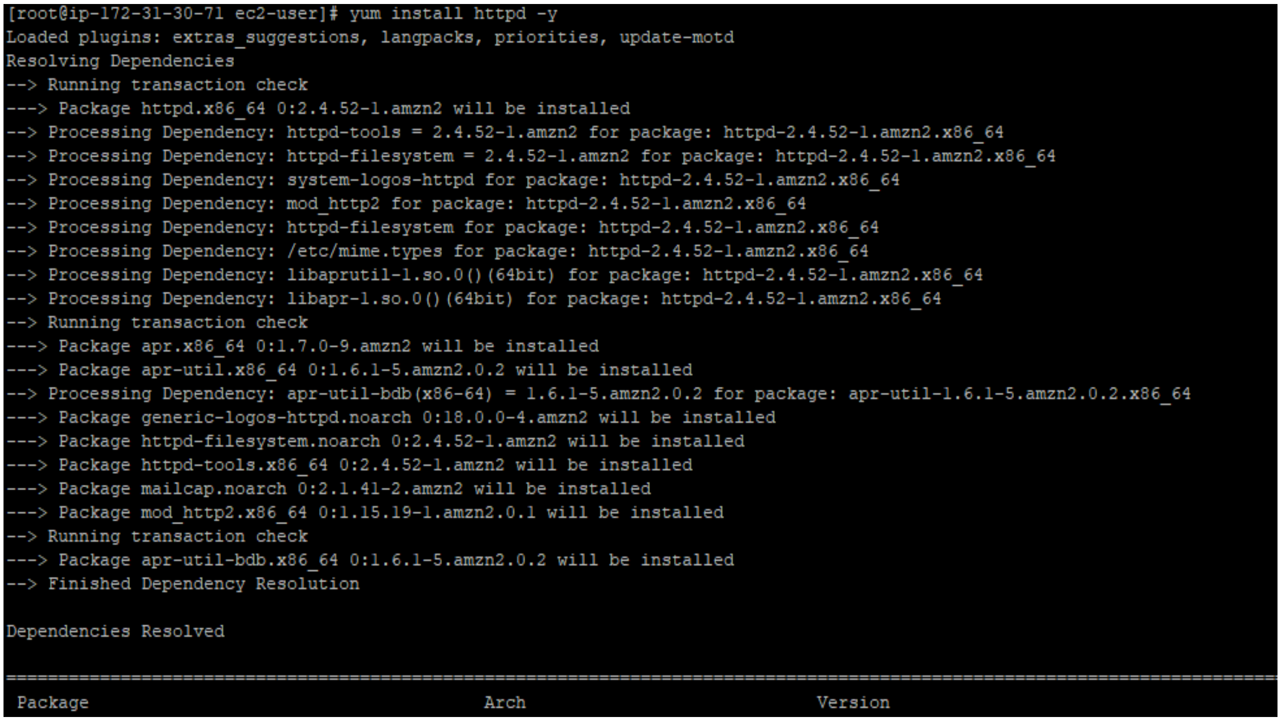
    3.5 **cd /var/www/html**

    3.6 **wget** (link of s3 zip file which have all the code of html and css)

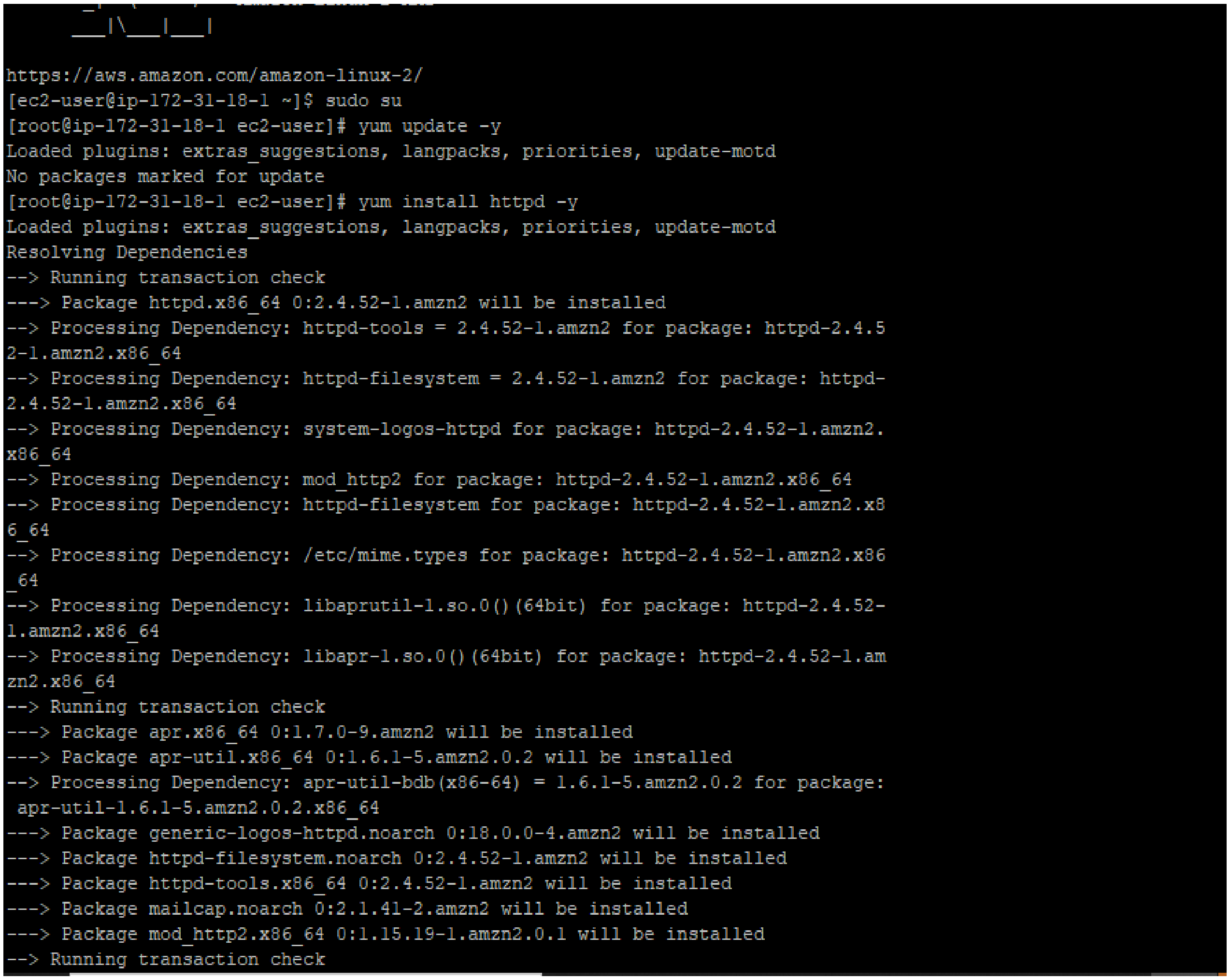
    3.7 **unzip file_name.zip(**to unzip the file in the instance)

    3.8 **ls(**to check all the files present in the folder)

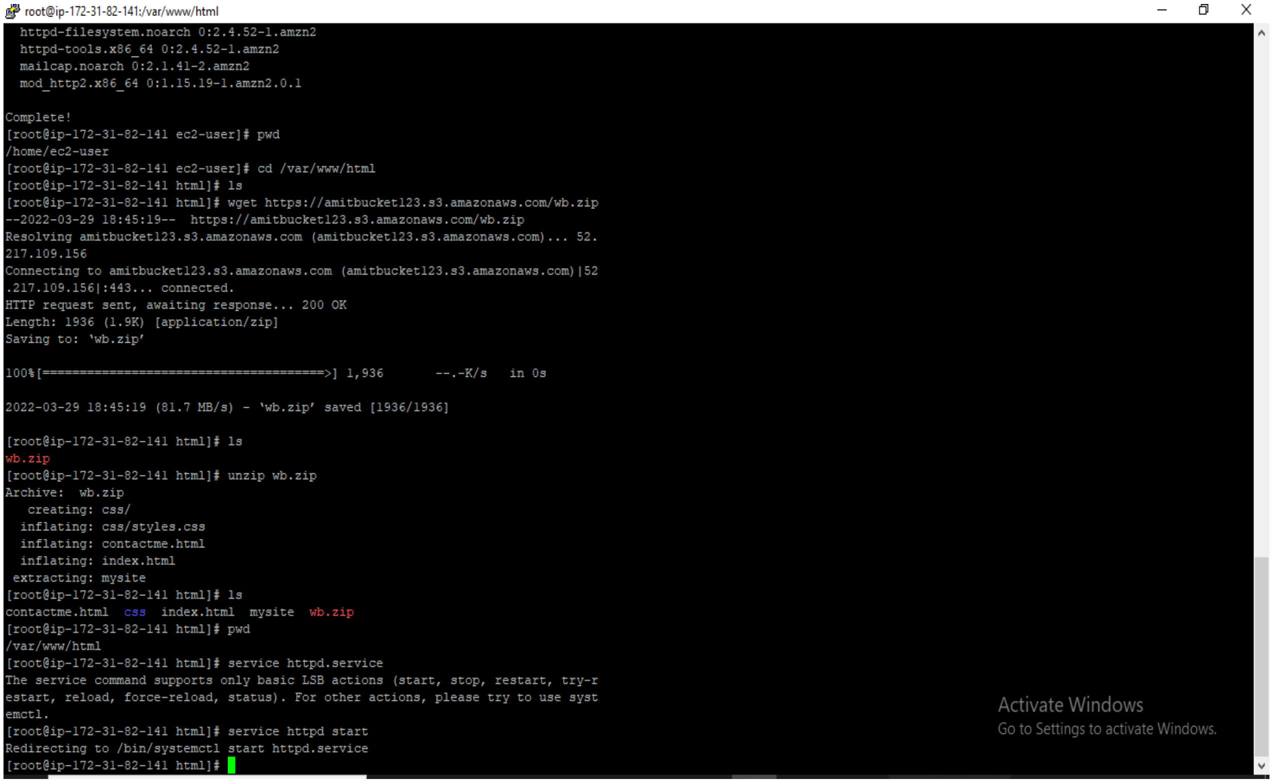
    3.9 **service httpd start(**to start the httpd service to host the website on the ec2 instance).



```
[root@ip-172-31-30-71 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.52-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.52-1.amzn2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.52-1.amzn2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.52-1.amzn2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.52-1.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.52-1.amzn2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==========================================================================================================================
 Package                                      Arch                            Version
```



```
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-18-1 ~]$ sudo su
[root@ip-172-31-18-1 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[root@ip-172-31-18-1 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.52-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.52-1.amzn2 for package: httpd-2.4.5
2-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.52-1.amzn2 for package: httpd-
2.4.52-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.52-1.amzn2.
x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.52-1.amzn2.x8
6_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.52-1.amzn2.x86
_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.52-
1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.52-1.am
zn2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package:
 apr-util-1.6.1-5.amzn2.0.2.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.52-1.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.52-1.amzn2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
```

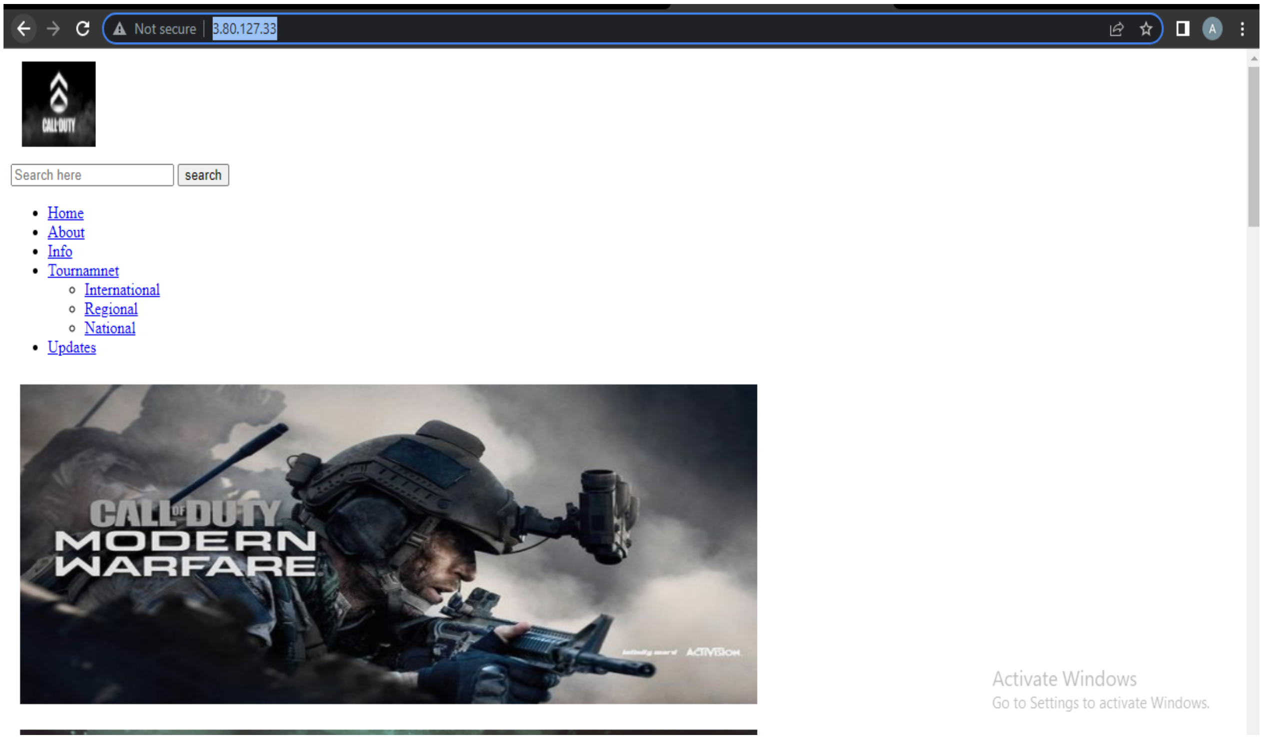


```
  httpd-filesystem.noarch 0:2.4.52-1.amzn2
  httpd-tools.x86_64 0:2.4.52-1.amzn2
  mailcap.noarch 0:2.1.41-2.amzn2
  mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[root@ip-172-31-82-141 ec2-user]# pwd
/home/ec2-user
[root@ip-172-31-82-141 ec2-user]# cd /var/www/html
[root@ip-172-31-82-141 html]# ls
[root@ip-172-31-82-141 html]# wget https://amitbucket123.s3.amazonaws.com/wb.zip
--2022-03-29 18:45:19--  https://amitbucket123.s3.amazonaws.com/wb.zip
Resolving amitbucket123.s3.amazonaws.com (amitbucket123.s3.amazonaws.com)... 52.
217.109.156
Connecting to amitbucket123.s3.amazonaws.com (amitbucket123.s3.amazonaws.com)|52
.217.109.156|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1936 (1.9K) [application/zip]
Saving to: ‘wb.zip’

100%[=====================================>] 1,936       --.-K/s   in 0s

2022-03-29 18:45:19 (81.7 MB/s) - ‘wb.zip’ saved [1936/1936]

[root@ip-172-31-82-141 html]# ls
wb.zip
[root@ip-172-31-82-141 html]# unzip wb.zip
Archive:  wb.zip
   creating: css/
  inflating: css/styles.css
  inflating: contactme.html
  inflating: index.html
 extracting: mysite
[root@ip-172-31-82-141 html]# ls
contactme.html  css  index.html  mysite  wb.zip
[root@ip-172-31-82-141 html]# pwd
/var/www/html
[root@ip-172-31-82-141 html]# service httpd.service
The service command supports only basic LSB actions (start, stop, restart, try-r
estart, reload, force-reload, status). For other actions, please try to use syst
emctl.
[root@ip-172-31-82-141 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-82-141 html]#
```

4. After successful start you get a message ‘Redirecting to start httpd.service

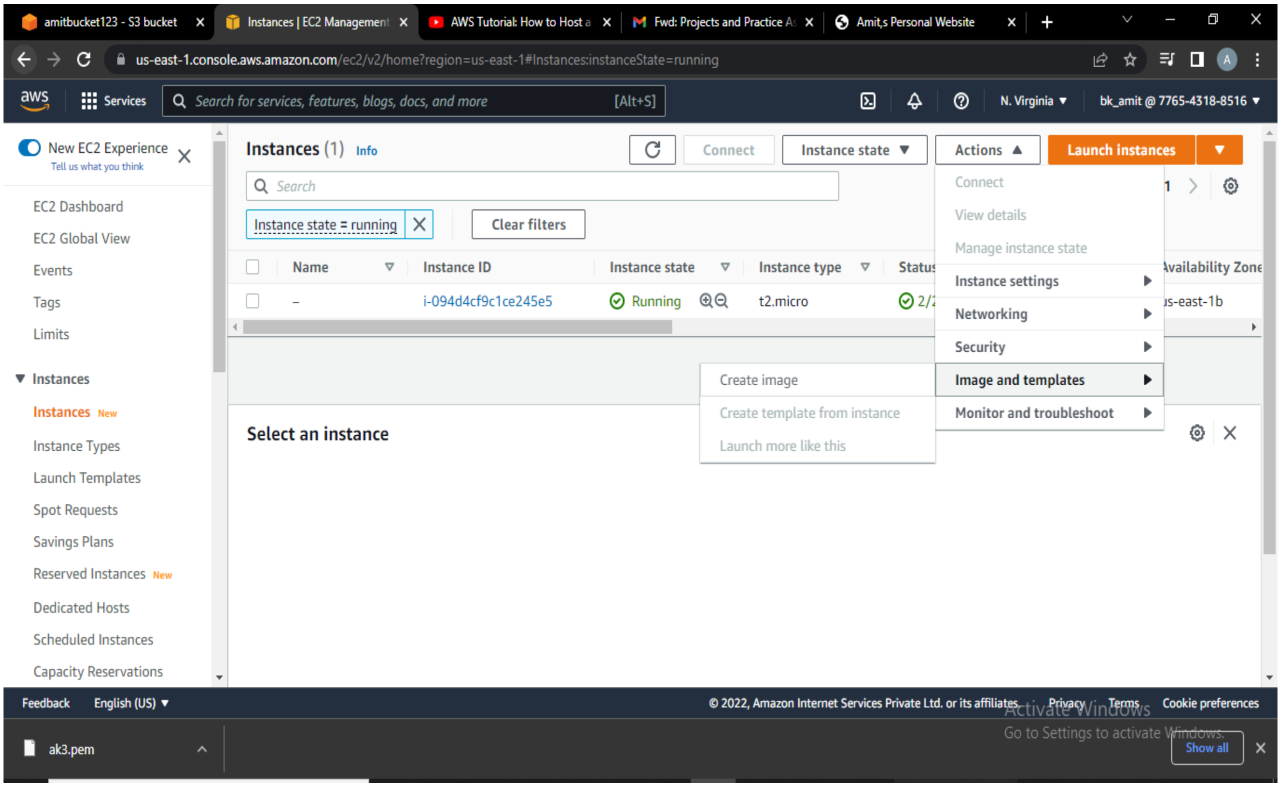
Now you can copy your instance public ipv4 address and paste in the browser and your website launched successfully on the ec2 instances.

Final snapshot of launched website on ec2 instance(linux)



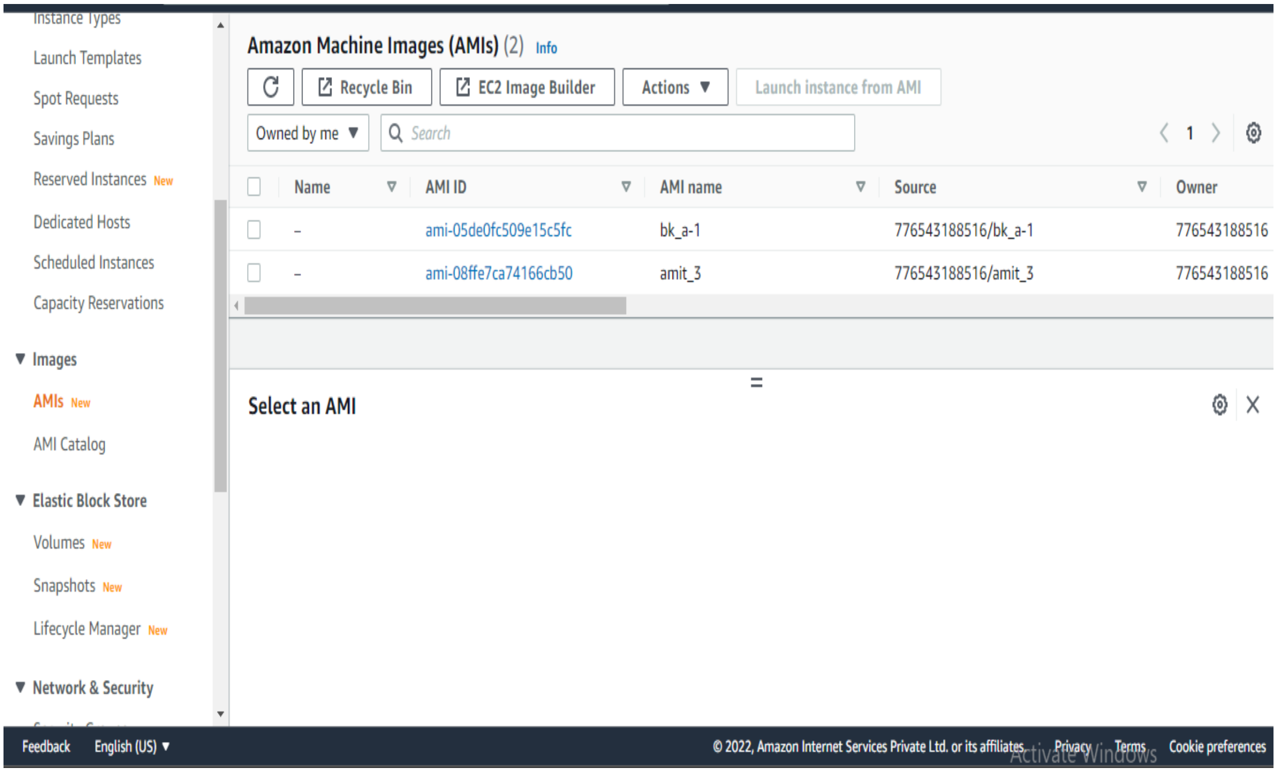
To create the AMI or snapshot of your instance.

1. Choose your instance for which you want to create and AMI(Amazon Machine images).Later on you can launch the instance of same configuration by choosing the option launch by AMI.



□	Name	Instance ID	Instance state	Instance type	Status	vailability Zone
□	–	i-094d4cf9c1ce245e5	Running	t2.micro	2/	is-east-1b

2. After creating the AMI you can seee your all AMI by clickin AMIs on the right side and later on you can launch instances by these AMIs.



Amazon Machine Images (AMIs) (2)

□	Name	AMI ID	AMI name	Source	Owner
□	–	ami-05de0fc509e15c5fc	bk_a-1	776543188516/bk_a-1	776543188516
□	–	ami-08ffe7ca74166cb50	amit_3	776543188516/amit_3	776543188516

Select an AMI