

# Advance AWS

## AWS Project- 1 (Day -5)

Student:

Kishore Shinde

Teacher:

Mrs. Vinolin Jeremiah

Course:

Advance AWS Cloud Computing with DevOps  
Fundamentals

Institute:

Lets Upgrade

# Project 01:

## Working with IAM Roles with S3 and Bootstrapping with EC2

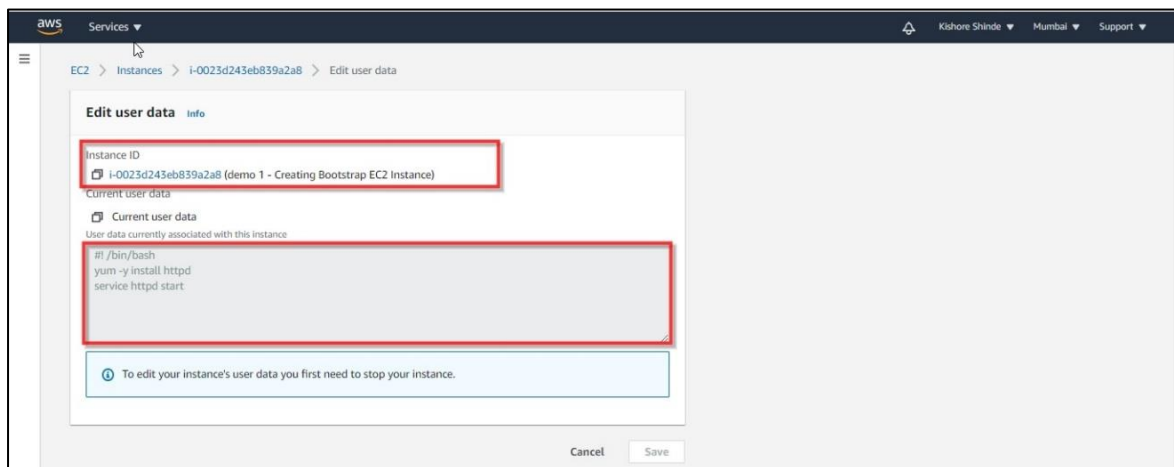
Task 1 : Creating a bootstrapped instance

Task 2 : Checking bucket list and creating a new bucket from EC2 using IAM Roles

Task 3 : Hosting a webpage using the bootstrap script on EC2

### Task 1: Creating a bootstrapped Instance

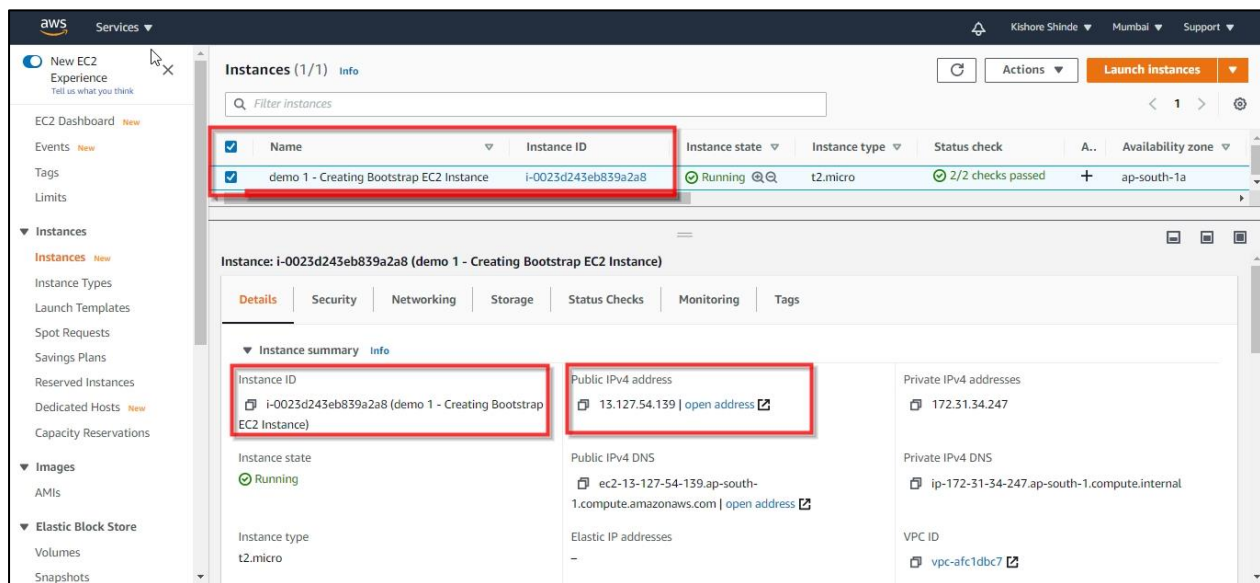
#### Screen 1 : Edit User Data



Script added in User Data:

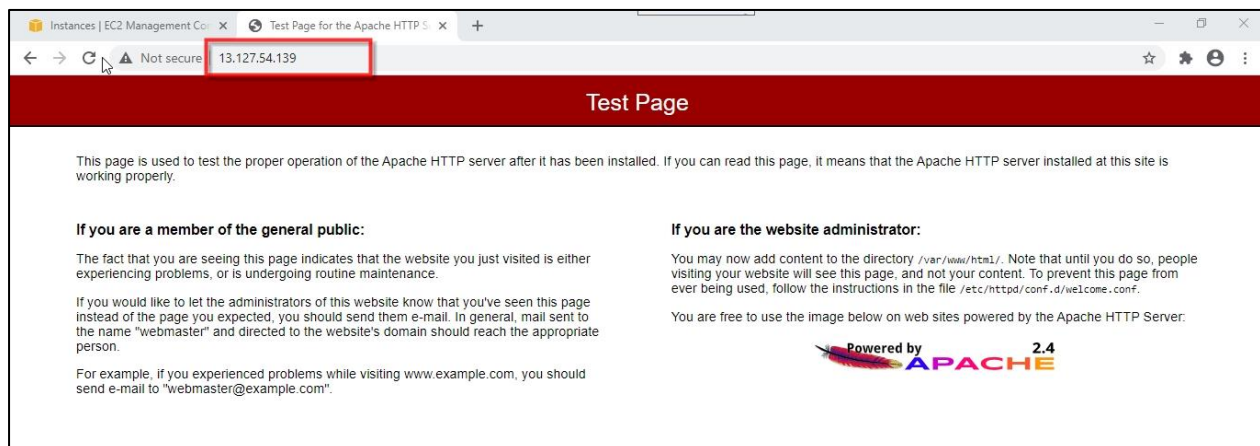
Sr. No.	Command	Description
1.	#!/bin/bash	Executes the script using the Bash shell
2.	yum -y install httpd	Installing Apache Software
3.	Service httpd start	Starts Apache Service

## Screen 2 : List of EC2 Instances



Instance Name	Instance ID	Public IP
demo 1 – Creating Bootstrap EC2 instance	i-0023d243eb839a2a8	13.127.54.139

## Screen 3 : Public IP Test Page



## Task 2 : Checking bucket list and creating a new bucket from EC2 using IAM Roles

### Screen 1 : User Data (IAM Role Selection : s3\_full\_access)

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-afc1dbc7 (default) [Create new VPC](#)

Subnet: No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP: Use subnet setting (Enable)

Placement group: ☐ Add instance to placement group

Capacity Reservation: Open

Domain join directory: No directory [Create new directory](#)

**IAM role: s3\_full\_access [Create new IAM role](#)**

Shutdown behavior: Stop

Stop - Hibernate behavior: ☐ Enable hibernation as an additional stop behavior

Enable termination protection: ☐ Protect against accidental termination

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

**Note :** No User Data added only role assigned in the above task.

### Screen 2 : List of EC2 instance with description

Instances (1/2) Info

Filter instances

Name	Instance ID	Instance state	Instance type	Status check	A..	Availability zone
demo 1 - Creating Bootstrap EC2 Instance	i-0023d243eb839a2a8	Running	t2.micro	2/2 checks passed	+	ap-south-1a
<b>demo 2 - EC2- Bucket Creation With S3 IAM Role</b>	<b>i-02d87993e2e96d8ee</b>	Running	t2.micro	2/2 checks passed	+	ap-south-1a

Instance: i-02d87993e2e96d8ee (demo 2 - EC2- Bucket Creation With S3 IAM Role)

Details | Security | Networking | Storage | Status Checks | Monitoring | Tags

Instance summary Info

Instance ID i-02d87993e2e96d8ee (demo 2 - EC2- Bucket Creation With S3 IAM Role)	Public IPv4 address 13.233.247.98   <a href="#">open address</a>	Private IPv4 addresses 172.31.37.169
Instance state Running	Public IPv4 DNS ec2-13-233-247-98.ap-south-1.compute.amazonaws.com   <a href="#">open address</a>	Private IPv4 DNS ip-172-31-37-169.ap-south-1.compute.internal
Instance type t2.micro	Elastic IP addresses -	VPC ID vpc-afc1dbc7

Instance Name	Instance ID	Public IP
demo 2 – EC2- Bucket Creation with S3 IAM Role	i-02d8799e2e96d8ee	13.233.247.98

### Screen 3 : 3 Commands executed

```
Last login: Wed Oct 14 08:52:08 2020 from ec2-13-233-177-0.ap-south-1.compute.amazonaws.com
[
  _l  ( _l  )
  _l  ( _l  )
  _l  ( _l  )
  _l  ( _l  )
] Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-37-169 ~]$ sudo su
[root@ip-172-31-37-169 ec2-user]# aws s3 ls
2020-10-14 08:51:46 s230106-01
2020-10-14 08:50:47 s230106-03
2020-10-14 08:22:13 s230106
2020-10-14 08:23:01 s230106-01
2020-10-14 08:23:25 s230106-02
[root@ip-172-31-37-169 ec2-user]# aws s3 mb s3://s230106-04
make_bucket: s230106-04
[root@ip-172-31-37-169 ec2-user]# aws s3 ls
2020-10-14 08:51:46 s230106-01
2020-10-14 08:50:47 s230106-03
2020-10-14 08:53:02 s230106-04
2020-10-14 08:22:13 s230106
2020-10-14 08:23:01 s230106-01
2020-10-14 08:23:25 s230106-02
[root@ip-172-31-37-169 ec2-user]# aws s3 mb s3://s230106-01
make_bucket: s230106-01
[root@ip-172-31-37-169 ec2-user]# aws s3 mb s3://s230106
make_bucket failed: s3://s230106 An error occurred (BucketAlreadyOwnedByYou) when calling the CreateBucket operation: Your previous request to create the named bucket
succeeded and you already own it.
[root@ip-172-31-37-169 ec2-user]#
```

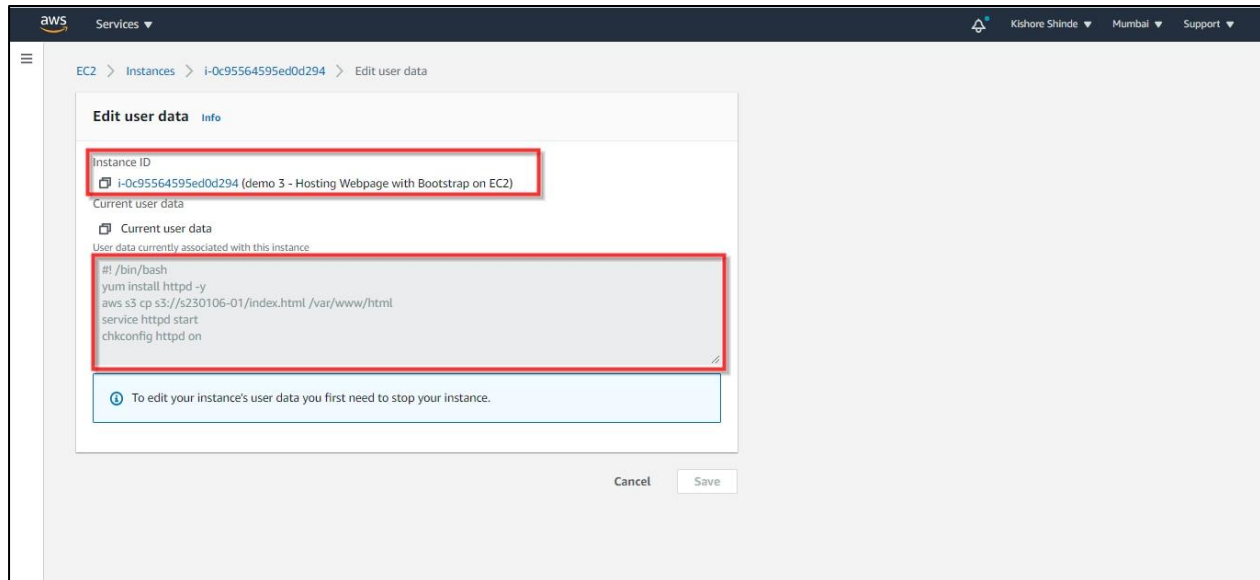
i-02d87993e2e96d8ee (demo 2 - EC2- Bucket Creation With S3 IAM Role)

Public IPs: 13.233.247.98    Private IPs: 172.31.37.169

Sr. No.	Command	Description
1.	aws s3 ls	List of files & folders in s3 bucket
2.	aws s3 mb s3://s230106-04	Create a bucket name i.e. s230106-04
3.	aws s3 mb s3://s230106-01	Will create a bucket with the name mentioned or If already exists will throw an error "Bucket already owned by you"

## Task 3 : Hosting a webpage using the bootstrap script on EC2

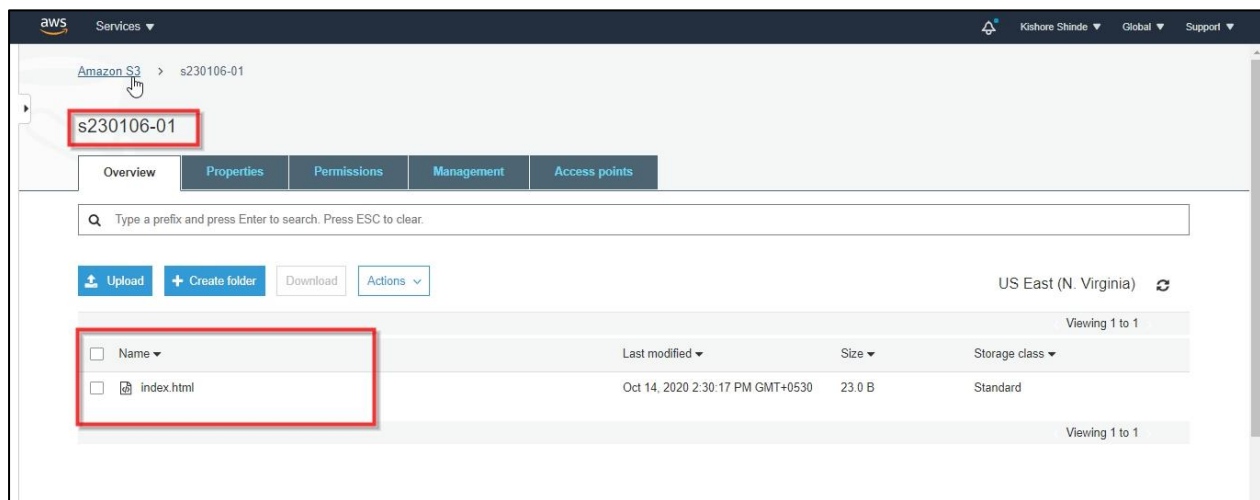
### Screen 1 : Edit User Data



### Script Added in User Data

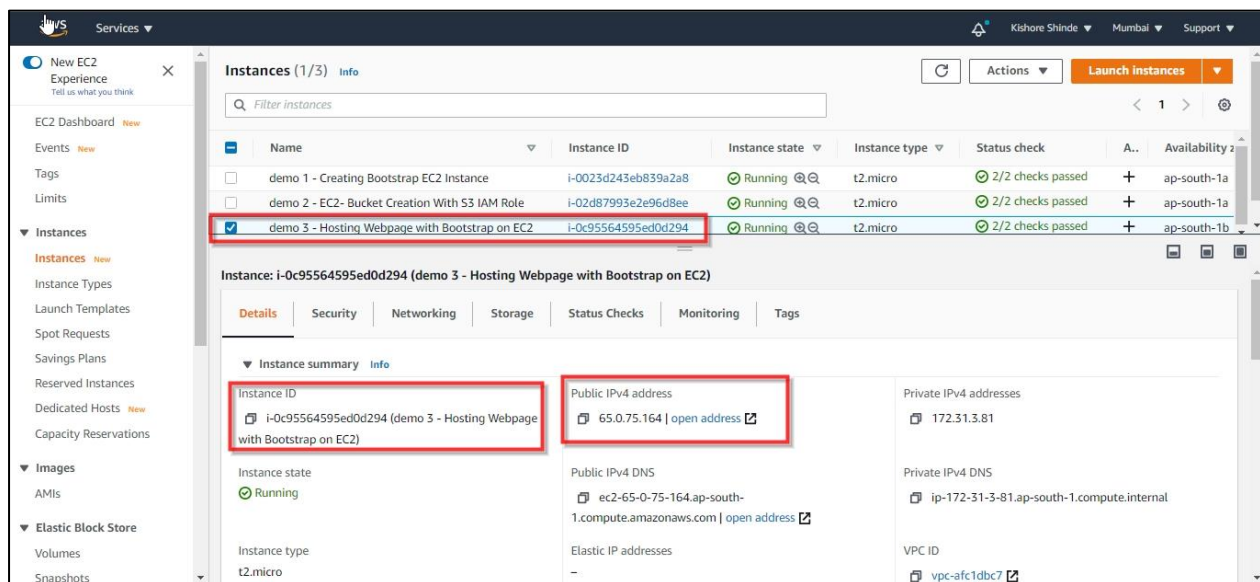
Sr. No.	Command	Description
1.	<code>#!/bin/bash</code>	Executes the script using the Bash shell
2.	<code>yum -y install httpd</code>	Installing Apache Software
3.	<code>aws s3 cp s3://s230106-01/index.html /var/www/html</code>	This command copies the index.html file from the given bucket to the mentioned path
4.	<code>Service httpd start</code>	Starts Apache Service
5.	<code>chkconfig httpd on</code>	Enables Apache Service

## Screen 2 : s3 bucket containing index.html



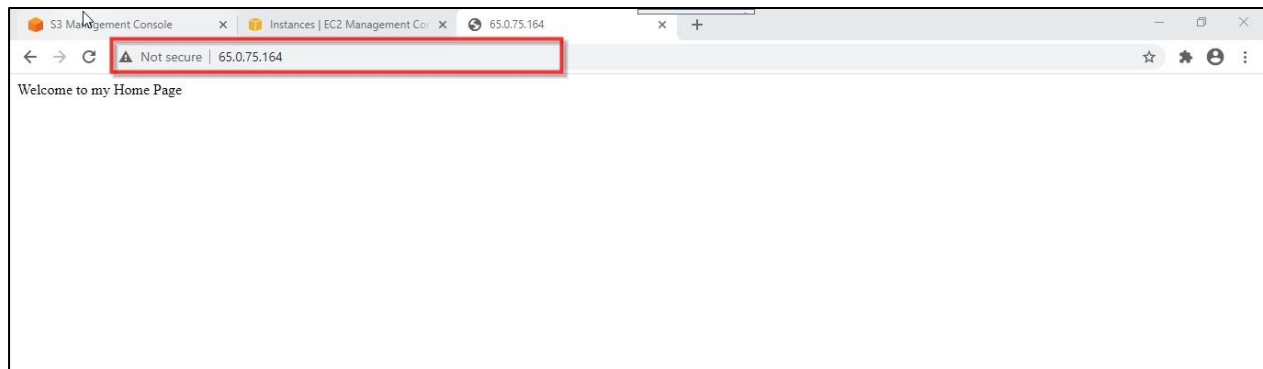
Bucket Name : s230106-01

## Screen 3 : List of EC 2 Instance with Description



Instance Name	Instance ID	Public IP
demo 3 – Hosting Webpage with Bootstrap on EC2	i-0c95564595ed0d294	65.0.75.164

## Screen 4 : Testing Public IP



Public IP : 65.0.75.164

Project 1 Ends Here



# Advance AWS

## AWS Project- 2 (Day -6)

Student:

Kishore Shinde

Teacher:

Mrs. Vinolin Jeremiah

Course:

Advance AWS Cloud Computing with DevOps  
Fundamentals

Institute:

Lets Upgrade

# Project 02:

## Creating and EC2 Instance in Custom VPC

Task 1 : Create VPC

Task 2 : Create an Internet Gateway

Task 3 : Create an Route Table

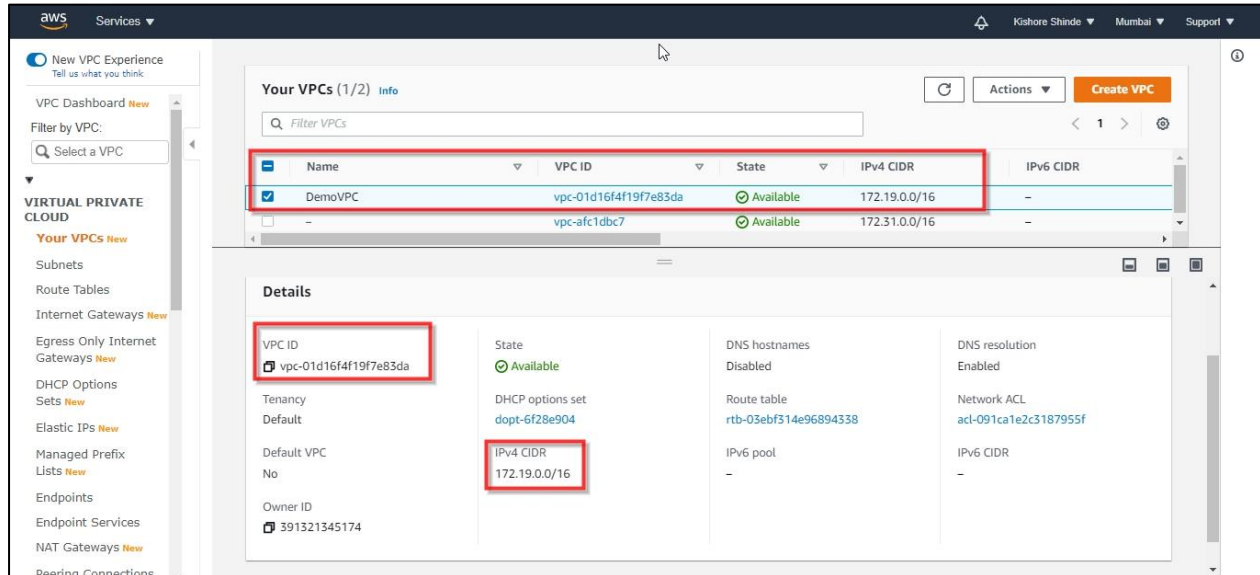
Task 4 : Create a Subnet

Task 5 : Create an EC2 in Custom VPC

Task 6 : Check ipconfig in VM command prompt

Task 1: Create VPC

Screenshot 1 : VPC Created

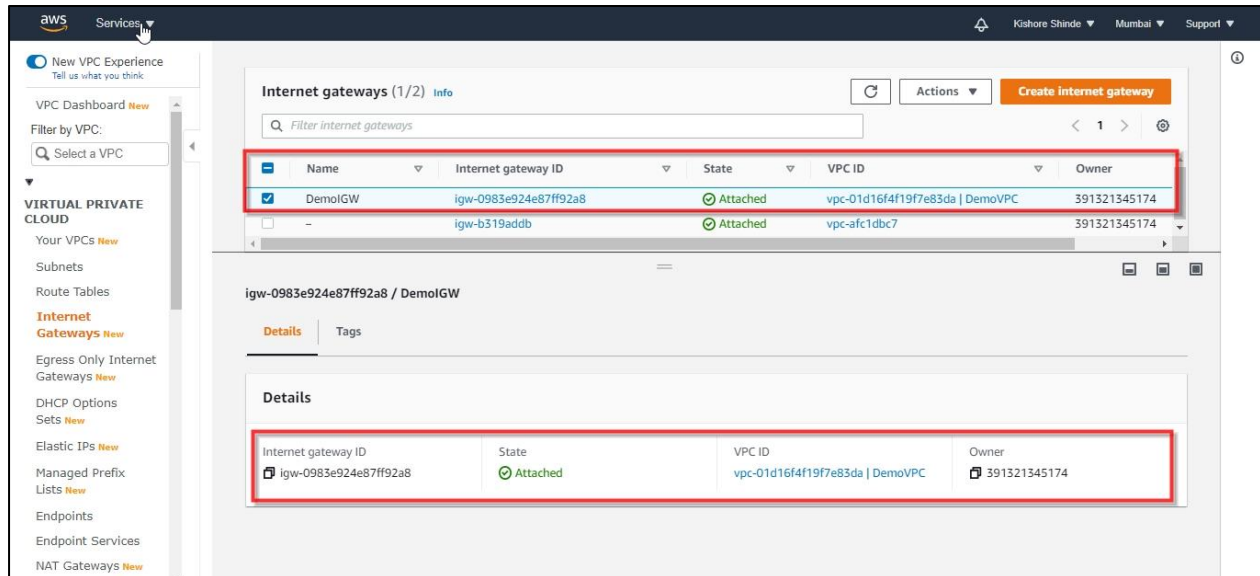


VPC Details:

Sr. No.	VPC Name	VPC ID	IPv4 CIDR
1.	DemoVPC	vpc-01d16f4f19f7e83da	172.19.0.0/16

## Task 2 : Create an Internet Gateway

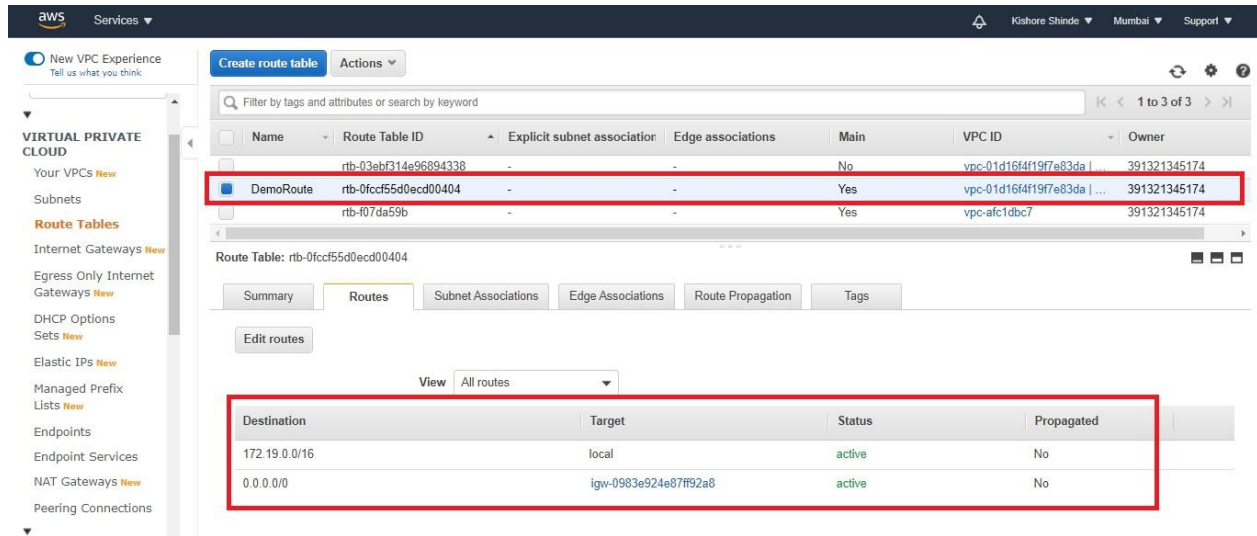
### Screenshot 2 : Internet Gateway Associated



Internet Gateway Name	Internet Gateway ID	VPC ID	State
DemoIGW	igw-0983e924e87ff92a8	vpc-01d16f4f19f7e83da (DemoVPC)	Attached

### Task 3 : Create Route Table

#### Screenshot 3 : Route Table with routes



Route Table Name	Route Table ID	VPC ID	Main
DemoIGW	rtb-0fccf55d0ecd00404	vpc-01d16f4f19f7e83da (DemoVPC)	Yes

#### All Route Details :

Destination	Target	Status
172.19.0.0/16	Local	active
0.0.0.0/16	igw-0983e924e87ff92a8 (Internet Gateway ID)	active

## Task 4 : Create a Subnet

### Screenshot 4 : Subnet Details

The screenshot shows the AWS Management Console for the 'VIRTUAL PRIVATE CLOUD' service. A table lists several subnets, with 'DemoSubnet01' (subnet-0623ab5d873357ad7) selected. The details for this subnet are shown below, including its VPC, state, CIDR blocks, and associated route table.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability
DemoSubnet01	subnet-0623ab5d873357ad7	available	vpc-01d16f4f19f7e83da   DemoVPC	172.19.19.0/24	251	-	ap-south-1b	aps1-az3
	subnet-11262579	available	vpc-afc1dbc7	172.31.32.0/20	4091	-	ap-south-1a	aps1-az1
	subnet-da384896	available	vpc-afc1dbc7	172.31.0.0/20	4091	-	ap-south-1b	aps1-az3
	subnet-fe259e85	available	vpc-afc1dbc7	172.31.16.0/20	4091	-	ap-south-1c	aps1-az2

Subnet: subnet-0623ab5d873357ad7

**Description** | Flow Logs | Route Table | Network ACL | Tags | Sharing

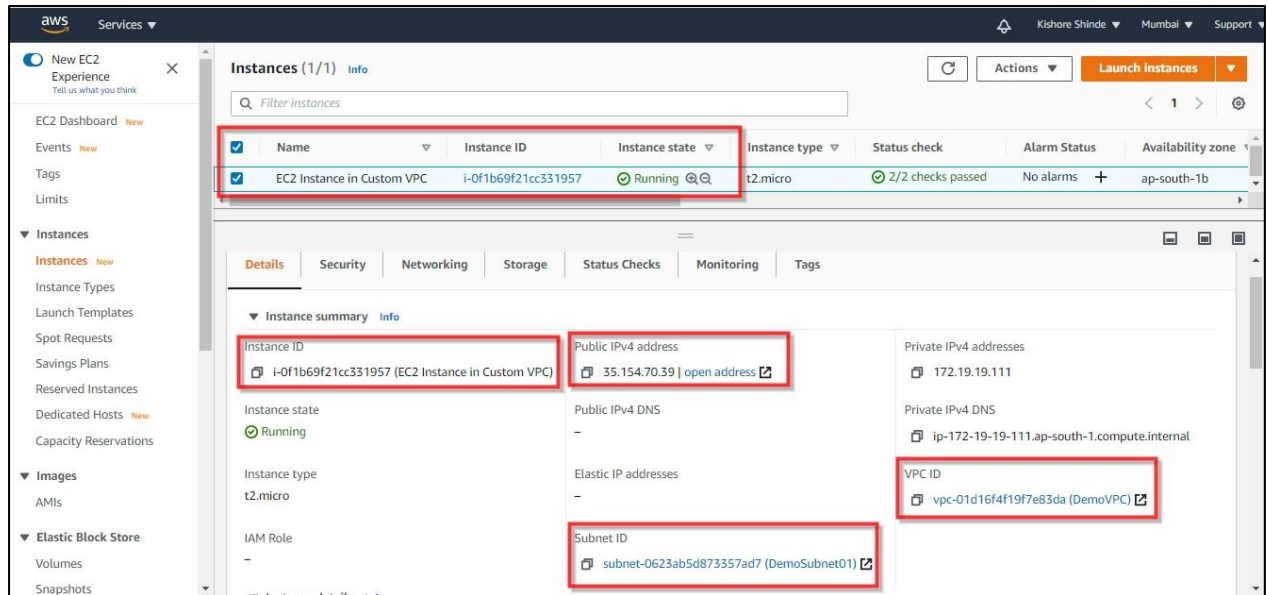
Subnet ID: subnet-0623ab5d873357ad7  
VPC: vpc-01d16f4f19f7e83da | DemoVPC  
State: available  
IPv4 CIDR: 172.19.19.0/24  
IPv6 CIDR: -  
Route Table: rtb-0fccf55d0ecd00404 | DemoRoute  
Default subnet: No  
Auto-assign customer-owned IPv4 address: No  
Auto-assign IPv6 address: No  
Owner: 391321345174

Available IPv4 Addresses: 251  
Availability Zone: ap-south-1b (aps1-az3)  
Network ACL: acl-091ca1e2c3187955f  
Auto-assign public IPv4 address: Yes  
Customer-owned IPv4 pool: -  
Outpost ID: -

Subnet Name	Subnet ID	VPC ID	Route Table	IPV4 CIDR
DemoSubnet01	subnet-0623ab5d873357ad7	vpc-01d16f4f19f7e83da (DemoVPC)	rtb-0fccf55d0ecd00404 (DemoRoute)	172.19.19.0/24

## Task 5 : Create an EC2 in Custom VPC

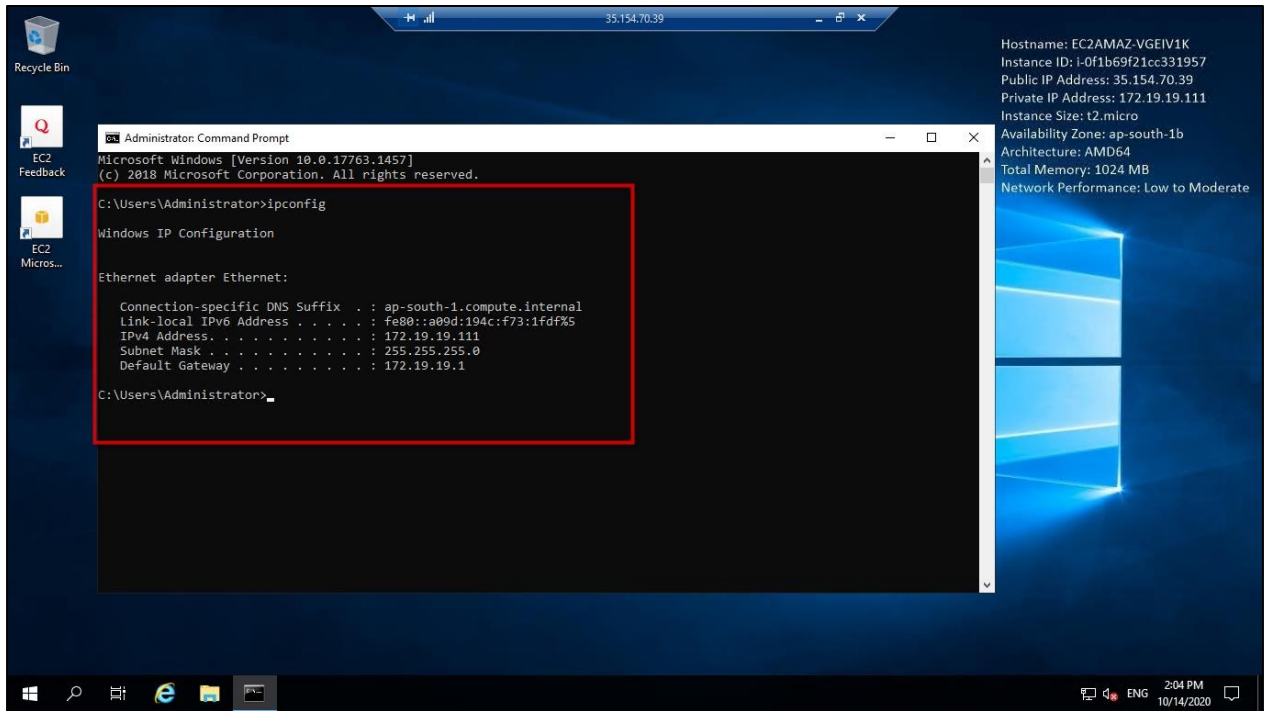
### Sreenshot 5 : EC2 Dashboard



Instance Name	Instance ID	Subnet ID	VPC ID	Public IPV4 Details
EC2 Instance in Custom VPC	i-0f1b69f21cc331957	subnet-0623ab5d873357ad7 (DemoSubnet01)	vpc-01d16f4f19f7e83da (DemoVPC)	35.154.70.39

## Task 6 : Check ipconfig in VM Command Prompt

### Screenshot 6 : cmd prompt:ipconfig



Public IPv4 Address	Private IP Address	Default Gateway
172.19.19.111	172.19.19.111	172.19.19.1

Project 2 Ends here