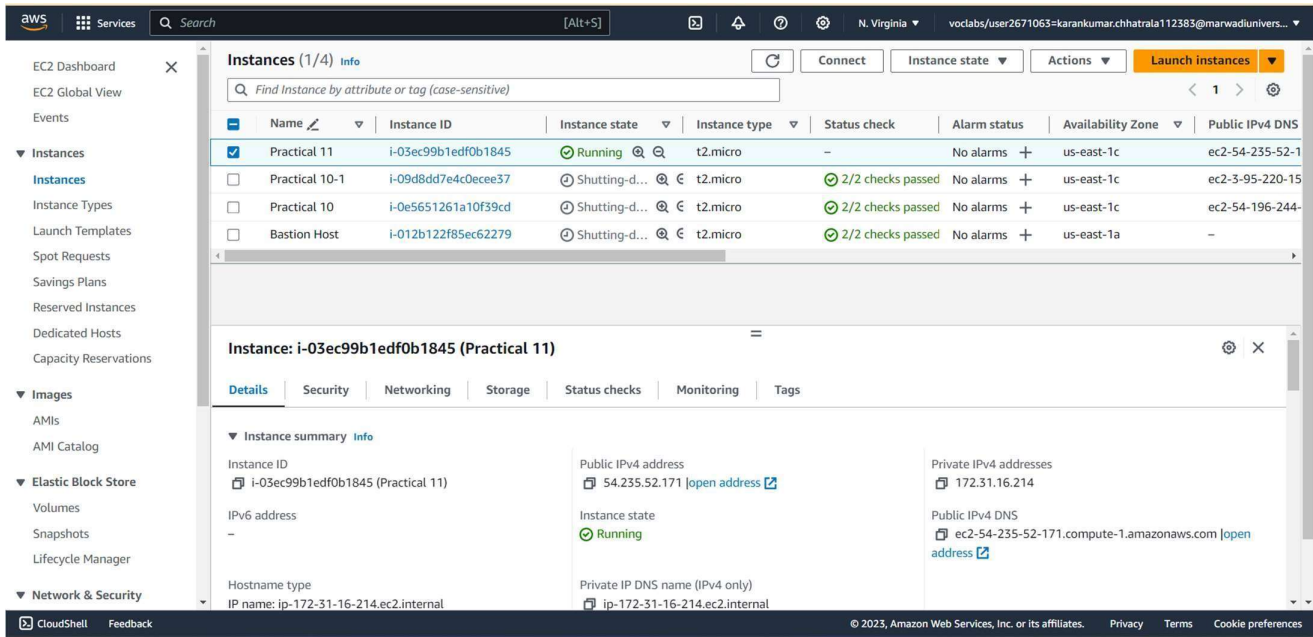


Practical 11 : Study on Hadoop Framework

Step 01 : Create an Instance in EC2. Go to Instance ID.

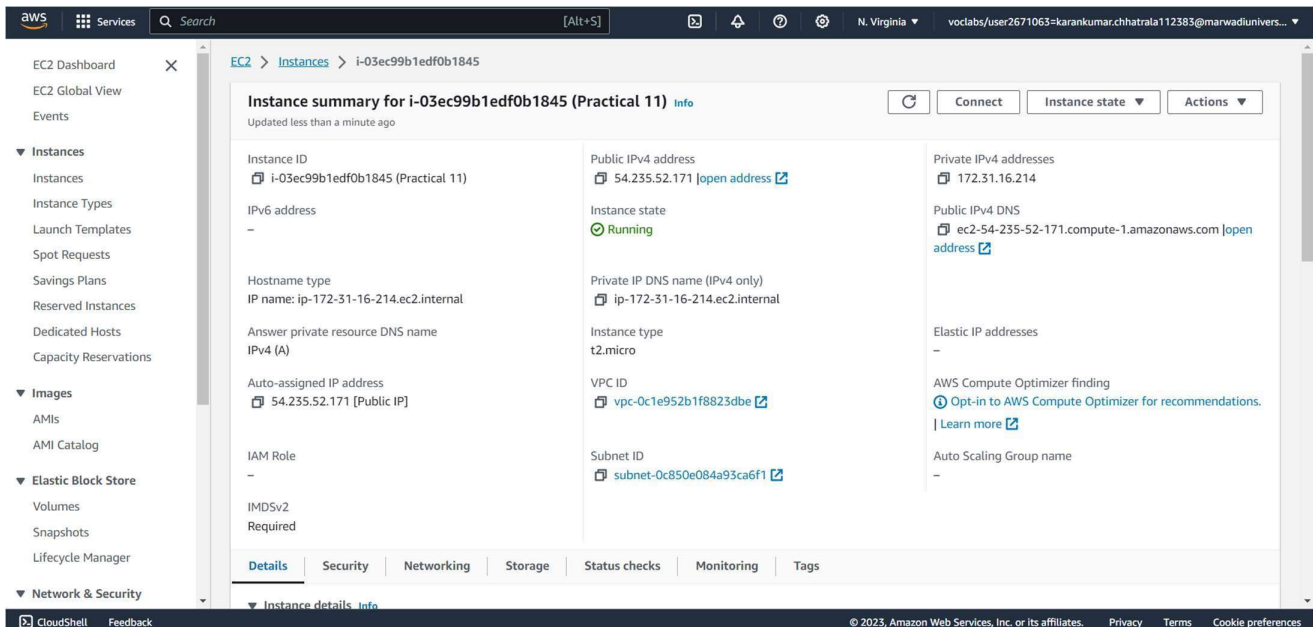
Snapshot :



The screenshot shows the AWS Management Console interface. On the left, the navigation menu includes 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The main content area displays a table of EC2 instances. The instance 'Practical 11' (ID: i-03ec99b1edf0b1845) is highlighted. Below the table, the 'Instance summary' for 'Practical 11' is shown, including details such as Instance ID, Public IPv4 address, Private IPv4 address, Instance state (Running), and Hostname type.

Step 02 : Click on Connect.

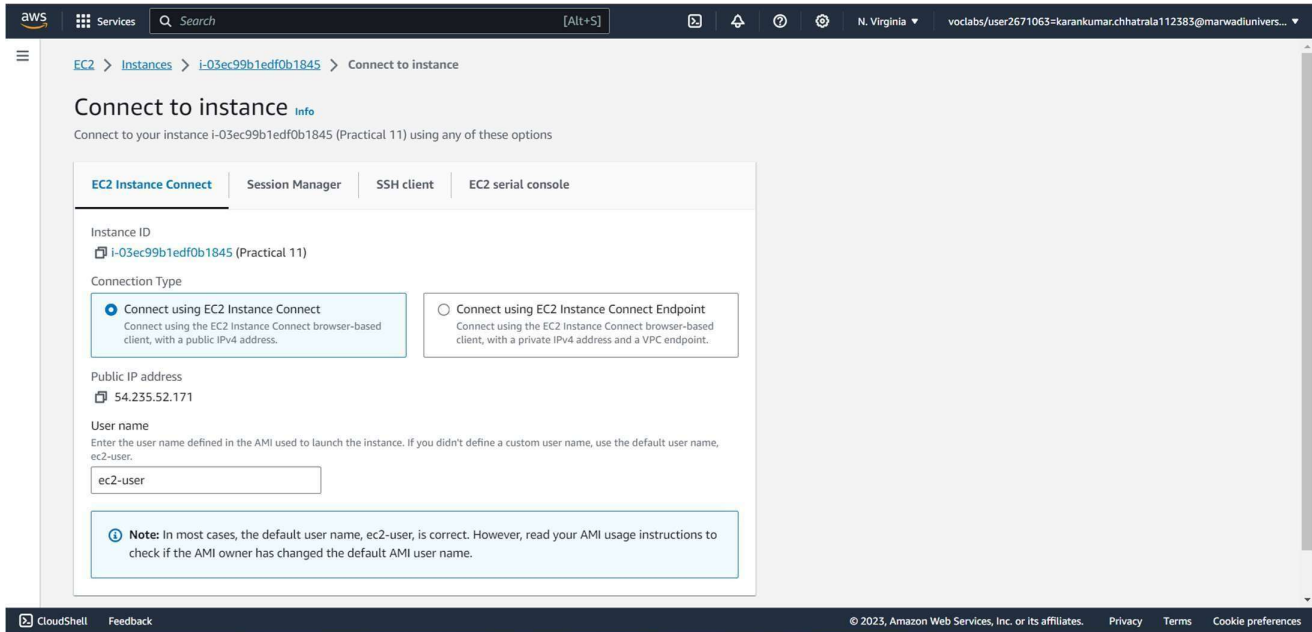
Snapshot :



The screenshot shows the 'Instance summary' page for 'Practical 11' (ID: i-03ec99b1edf0b1845). The page displays various details about the instance, including its Instance ID, Public IPv4 address (54.235.52.171), Private IPv4 address (172.31.16.214), Instance state (Running), Hostname type, VPC ID, and Subnet ID. The 'Connect' button is visible at the top right of the summary card.

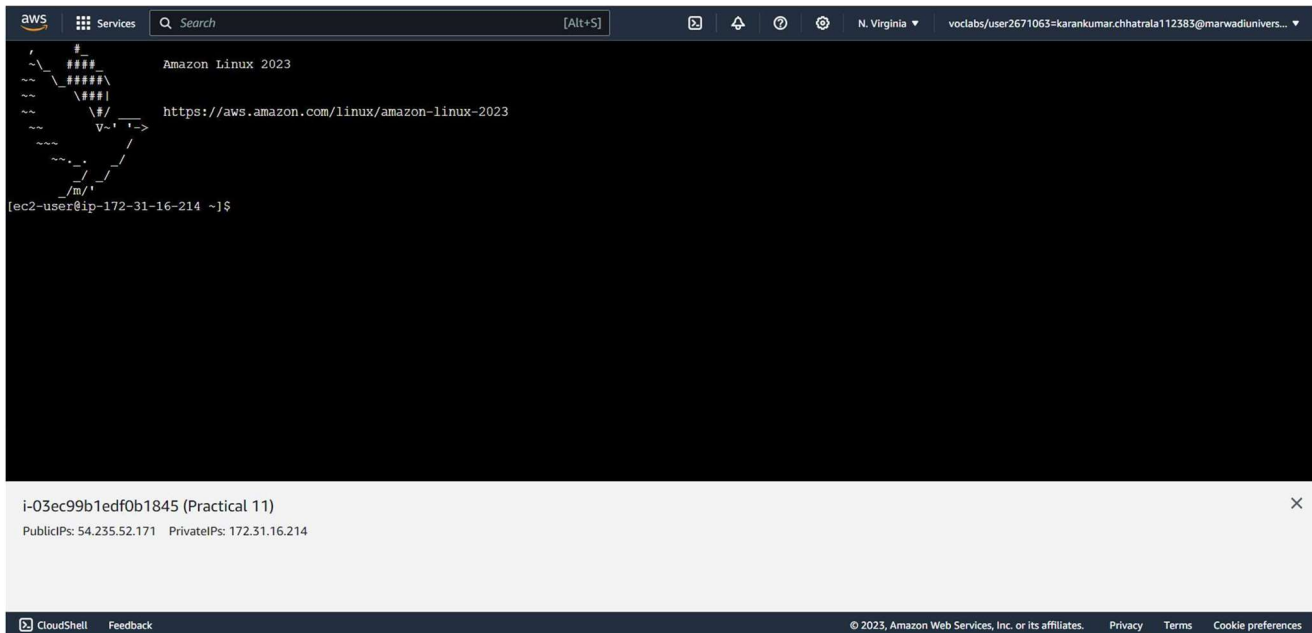
Step 03 : Click on Connect.

Snapshot :



Step 04 : An Instance in EC2 is created.

Snapshot :



Step 05 :Run command ‘sudo su -’.

Snapshot :

```
aws Services Search [Alt+S] N. Virginia voclabs/user2671063=karakumar.chhatrala112383@marwadiunivers...
[ec2-user@ip-172-31-16-214 ~]$ sudo su -
[root@ip-172-31-16-214 ~]#
```

Step 06 : Run command ‘yum update -y’.

Snapshot :

```
aws Services Search [Alt+S] N. Virginia voclabs/user2671063=karakumar.chhatrala112383@marwadiunivers...
[root@ip-172-31-16-214 ~]# yum update -y
Last metadata expiration check: 0:02:23 ago on Sat Oct 28 13:02:57 2023.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-16-214 ~]#
```

Step 07 : Run command ‘yum install nginx -y’.

Snapshot :

```
aws Services Search [Alt+S] N. Virginia voclabs/user2671063=karakumar.chhatrala112383@marwadiunivers...
[root@ip-172-31-16-214 ~]# yum install nginx -y
Last metadata expiration check: 0:02:48 ago on Sat Oct 28 13:02:57 2023.
Dependencies resolved.
=====
Package                                Architecture  Version                                Repository  Size
=====
Installing:
nginx                                  x86_64        1:1.24.0-1.amzn2023.0.2              amazonlinux 32 k
Installing dependencies:
generic-logos-httpd                  noarch        18.0.0-12.amzn2023.0.3              amazonlinux 19 k
gperftools-libs                      x86_64        2.9.1-1.amzn2023.0.2                amazonlinux 309 k
libunwind                            x86_64        1.4.0-5.amzn2023.0.2                amazonlinux 66 k
nginx-core                           x86_64        1:1.24.0-1.amzn2023.0.2              amazonlinux 586 k
nginxfilesystem                      noarch        1:1.24.0-1.amzn2023.0.2              amazonlinux 9.1 k
nginx-mimetypes                      noarch        2.1.49-3.amzn2023.0.3              amazonlinux 21 k
=====
Transaction Summary
-----
Install 7 Packages

Total download size: 1.0 M
Installed size: 3.4 M
Downloading Packages:
(1/7): nginx-core-1.24.0-1.amzn2023.0.2.x86_64.rpm              7.9 MB/s | 586 kB  00:00
(2/7): gperftools-libs-2.9.1-1.amzn2023.0.2.x86_64.rpm         3.8 MB/s | 309 kB  00:00
(3/7): nginx-1.24.0-1.amzn2023.0.2.x86_64.rpm                 383 kB/s | 32 kB   00:00
(4/7): libunwind-1.4.0-5.amzn2023.0.2.x86_64.rpm               4.5 MB/s | 66 kB   00:00
(5/7): nginxfilesystem-1.24.0-1.amzn2023.0.2.noarch.rpm        611 kB/s | 9.1 kB  00:00
-----
i-03ec99b1edf0b1845 (Practical 11)
PublicIPs: 54.235.52.171 PrivateIPs: 172.31.16.214

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```

AWS Services Search [Alt+S] N. Virginia voclabs/user2671063=karankumar.chhatrala112383@marwadiunivers...
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing :
  Running scriptlet: nginx-filessystem-1:1.24.0-1.amzn2023.0.2.noarch 1/1
  Installing : nginx-filessystem-1:1.24.0-1.amzn2023.0.2.noarch 1/7
  Installing : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch 1/7
  Installing : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 2/7
  Installing : libunwind-1.4.0-5.amzn2023.0.2.x86_64 3/7
  Installing : gperftools-libs-2.9.1-1.amzn2023.0.2.x86_64 4/7
  Installing : nginx-core-1:1.24.0-1.amzn2023.0.2.x86_64 5/7
  Installing : nginx-1:1.24.0-1.amzn2023.0.2.x86_64 6/7
  Running scriptlet: nginx-1:1.24.0-1.amzn2023.0.2.x86_64 7/7
  Verifying : nginx-1:1.24.0-1.amzn2023.0.2.x86_64 7/7
  Verifying : gperftools-libs-2.9.1-1.amzn2023.0.2.x86_64 1/7
  Verifying : nginx-core-1:1.24.0-1.amzn2023.0.2.x86_64 2/7
  Verifying : libunwind-1.4.0-5.amzn2023.0.2.x86_64 3/7
  Verifying : nginx-filessystem-1:1.24.0-1.amzn2023.0.2.noarch 4/7
  Verifying : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch 5/7
  Verifying : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 6/7
Installed:
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch gperftools-libs-2.9.1-1.amzn2023.0.2.x86_64 libunwind-1.4.0-5.amzn2023.0.2.x86_64
nginx-1:1.24.0-1.amzn2023.0.2.x86_64 nginx-core-1:1.24.0-1.amzn2023.0.2.x86_64 nginx-filessystem-1:1.24.0-1.amzn2023.0.2.noarch
nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch
Complete!
[root@ip-172-31-16-214 ~]#

i-03ec99b1edf0b1845 (Practical 11)
PublicIPs: 54.235.52.171 PrivateIPs: 172.31.16.214
CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

Step 08 : Run command ‘sudo service nginx start’.

Snapshot :

```

[root@ip-172-31-16-214 ~]# sudo service nginx start
Redirecting to /bin/systemctl start nginx.service
[root@ip-172-31-16-214 ~]#

```

Step 09 : Run command ‘systemctl start nginx.service’.

Snapshot :

```

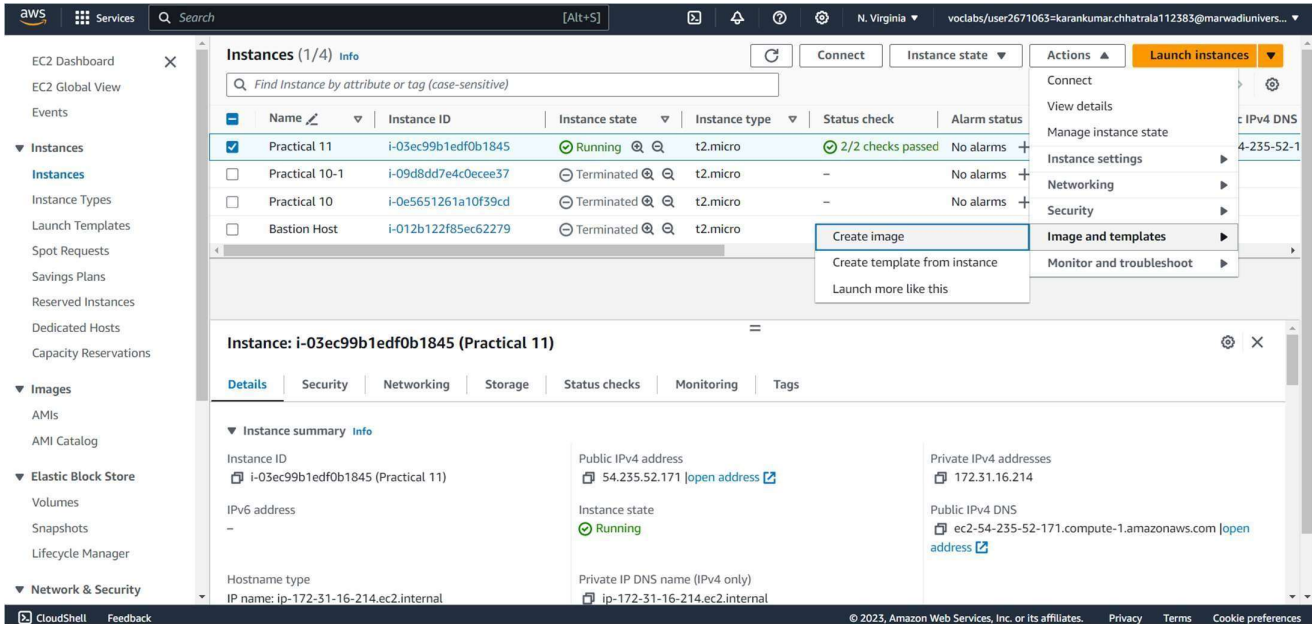
[root@ip-172-31-87-223 ~]# sudo service nginx start
Redirecting to /bin/systemctl start nginx.service
[root@ip-172-31-87-223 ~]# systemctl status nginx.service
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset: disabled)
   Active: active (running) since Fri 2023-09-29 02:25:17 UTC; 41s ago
     Process: 25976 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
     Process: 25977 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
     Process: 25978 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
    Main PID: 25979 (nginx)
      Tasks: 2 (limit: 1114)
     Memory: 2.2M
           CPU: 56ms
   CGroup: /system.slice/nginx.service
           └─25979 *nginx: master process /usr/sbin/nginx*
             └─25980 *nginx: worker process*

Sep 29 02:25:16 ip-172-31-87-223.ec2.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Sep 29 02:25:16 ip-172-31-87-223.ec2.internal nginx[25977]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Sep 29 02:25:16 ip-172-31-87-223.ec2.internal nginx[25977]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Sep 29 02:25:17 ip-172-31-87-223.ec2.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[root@ip-172-31-87-223 ~]#

```

Step 10 : Go to Instances >> Actions >> Image and Templates >> Create image.

Snapshot :



The screenshot shows the AWS Management Console interface. On the left, the navigation menu includes 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The main content area displays a list of instances. The 'Actions' menu for the selected instance 'Practical 11' is open, showing options like 'Connect', 'View details', 'Manage instance state', 'Instance settings', 'Networking', 'Security', 'Image and templates', and 'Monitor and troubleshoot'. The 'Create image' option is highlighted. Below this, the details for instance 'i-03ec99b1edf0b1845 (Practical 11)' are shown, including its ID, state (Running), type (t2.micro), and various addresses.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
Practical 11	i-03ec99b1edf0b1845	Running	t2.micro	2/2 checks passed	No alarms
Practical 10-1	i-09d8dd7e4c0ecce37	Terminated	t2.micro	-	No alarms
Practical 10	i-0e5651261a10f39cd	Terminated	t2.micro	-	No alarms
Bastion Host	i-012b122f85ec62279	Terminated	t2.micro	-	No alarms

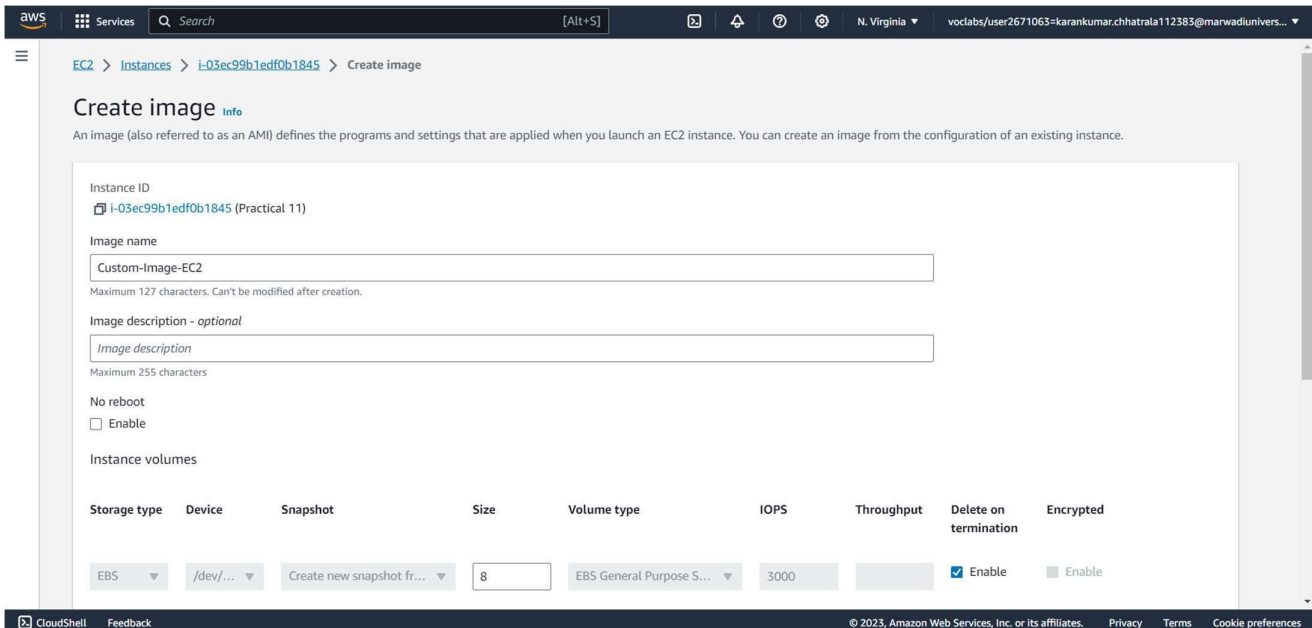
Instance: i-03ec99b1edf0b1845 (Practical 11)

Instance summary

- Instance ID: i-03ec99b1edf0b1845 (Practical 11)
- IPv6 address: -
- Hostname type: IP name: ip-172-31-16-214.ec2.internal
- Public IPv4 address: 54.235.52.171 [open address]
- Instance state: Running
- Private IP DNS name (IPv4 only): ip-172-31-16-214.ec2.internal
- Private IPv4 addresses: 172.31.16.214
- Public IPv4 DNS: ec2-54-235-52-171.compute-1.amazonaws.com [open address]

Step 11 : Create image.

Snapshot :



The screenshot shows the 'Create image' page in the AWS Management Console. The page title is 'Create image'. Below the title, there is a description: 'An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.' The form includes the following fields:

- Instance ID:** i-03ec99b1edf0b1845 (Practical 11)
- Image name:** Custom-Image-EC2
- Image description - optional:** Image description
- No reboot:** ☐ Enable
- Instance volumes:** A table with columns: Storage type, Device, Snapshot, Size, Volume type, IOPS, Throughput, Delete on termination, and Encrypted.

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/...	Create new snapshot fr...	8	EBS General Purpose S...	3000		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

Step 12 : Select ‘Tag image and snapshots together.’ And click on Create image.

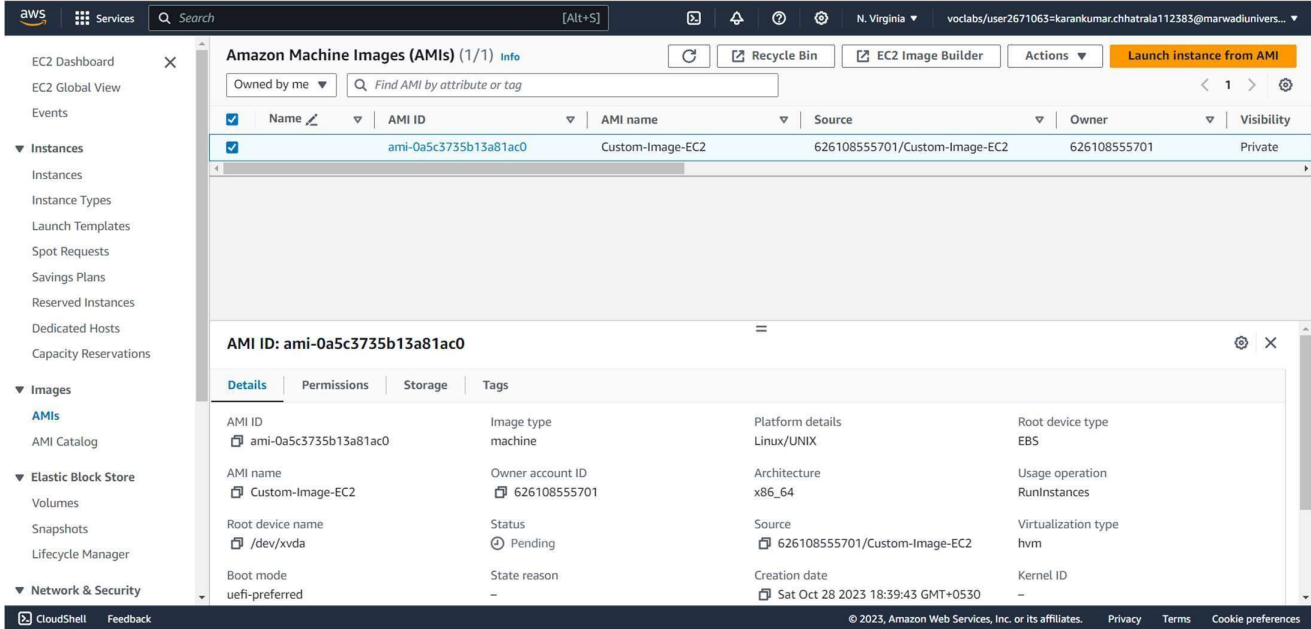
Snapshot :

Step 13 : AMI created.

Snapshot :

Step 14 : Go to Images >> AMIs >> AMI ID

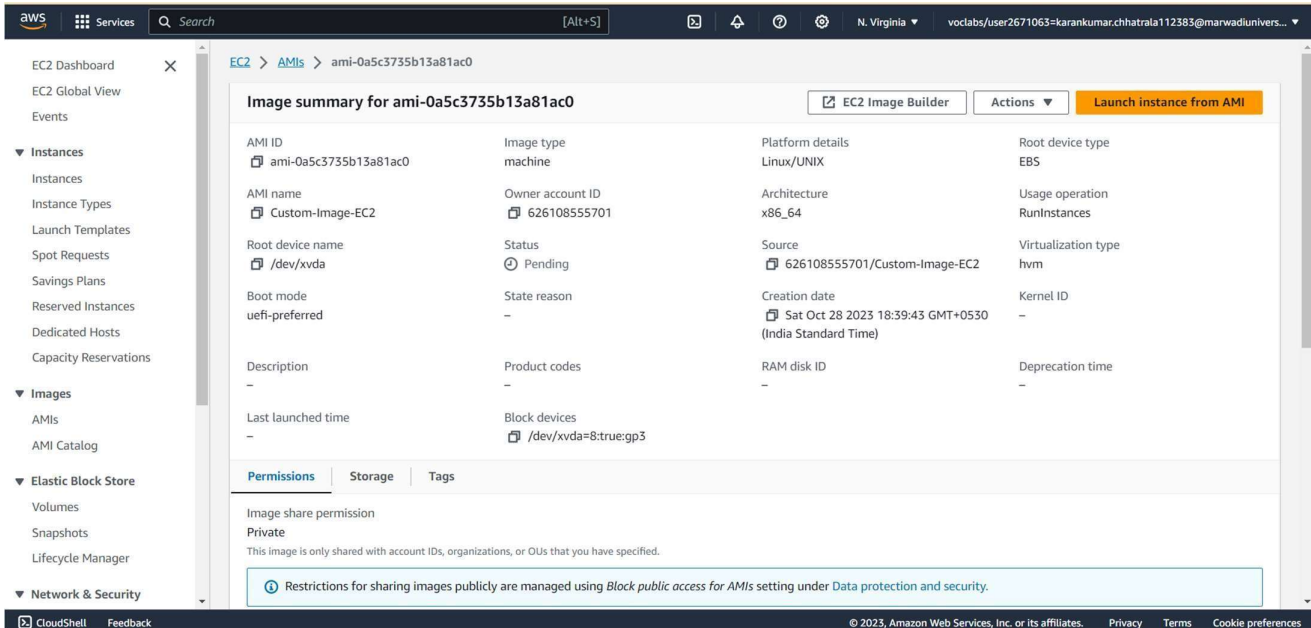
Snapshot :



The screenshot shows the AWS Management Console interface. On the left, the navigation menu is visible with categories like EC2 Dashboard, Elastic Block Store, and Network & Security. The main content area displays the 'Amazon Machine Images (AMIs) (1/1)' page. A table lists the AMIs, with one entry selected: ami-0a5c3735b13a81ac0, Custom-Image-EC2, 626108555701/Custom-Image-EC2, 626108555701, Private. Below the table, the 'Details' tab for the selected AMI is shown, providing information such as AMI ID, AMI name, Root device name, Boot mode, Image type, Owner account ID, Status, Platform details, Architecture, Source, Creation date, Root device type, Usage operation, Virtualization type, and Kernel ID.

Step 15 : Launch instance from AMI. Create 3 Instances.

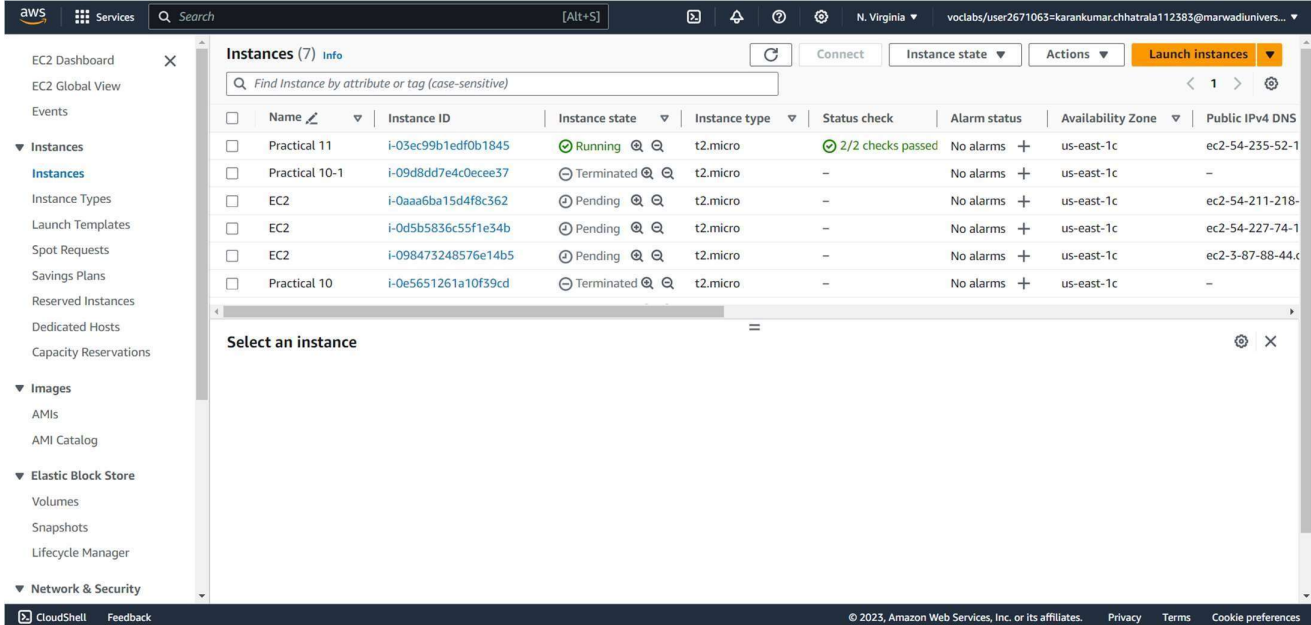
Snapshot :



The screenshot shows the AWS Management Console interface, specifically the 'Image summary' page for the AMI ami-0a5c3735b13a81ac0. The page displays a detailed overview of the AMI, including its ID, name, root device name, boot mode, image type, owner account ID, status, platform details, architecture, source, creation date, root device type, usage operation, virtualization type, and kernel ID. Below the summary, the 'Permissions' tab is selected, showing the image share permission as 'Private' and a note about restrictions for sharing images publicly.

Step 16 : Instances created.

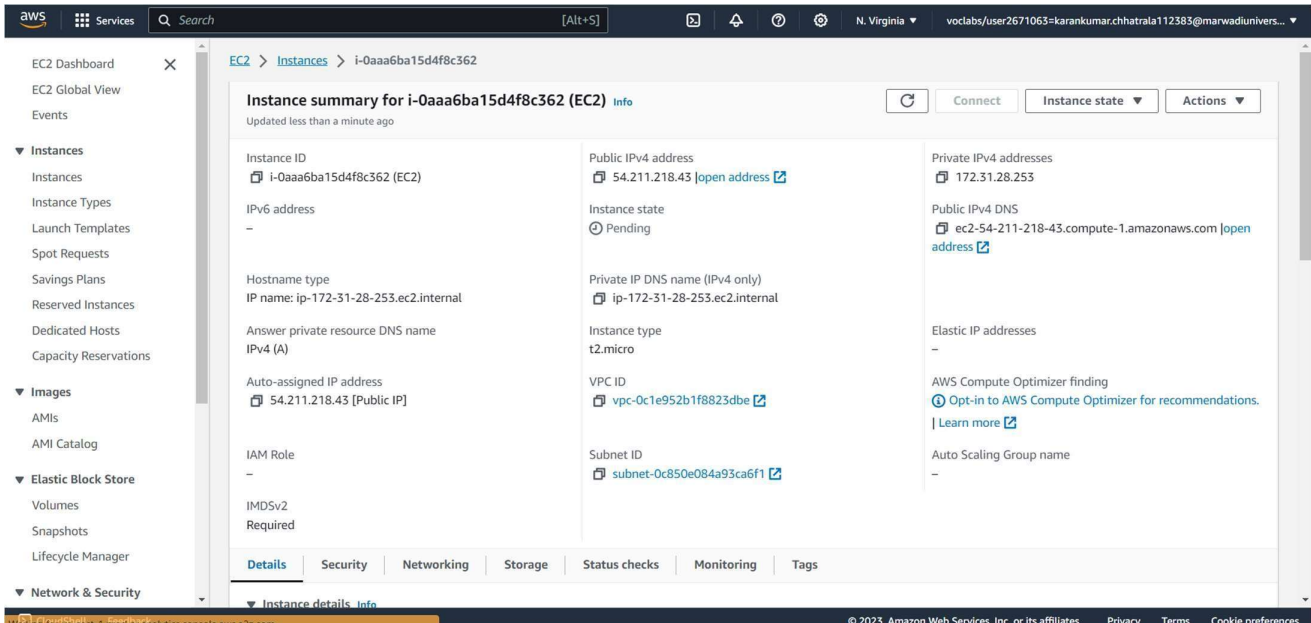
Snapshot :



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Practical 11	i-03ec99b1edf0b1845	Running	t2.micro	2/2 checks passed	No alarms	us-east-1c	ec2-54-235-52-1
Practical 10-1	i-09d8dd7e4c0ecee37	Terminated	t2.micro	-	No alarms	us-east-1c	-
EC2	i-0aaa6ba15d4f8c362	Pending	t2.micro	-	No alarms	us-east-1c	ec2-54-211-218-
EC2	i-0d5b5836c5f1e34b	Pending	t2.micro	-	No alarms	us-east-1c	ec2-54-227-74-1
EC2	i-098473248576e14b5	Pending	t2.micro	-	No alarms	us-east-1c	ec2-3-87-88-44.c
Practical 10	i-0e5651261a10f39cd	Terminated	t2.micro	-	No alarms	us-east-1c	-

Step 17 : Connect Instance 1.

Snapshot :



Instance summary for i-0aaa6ba15d4f8c362 (EC2)

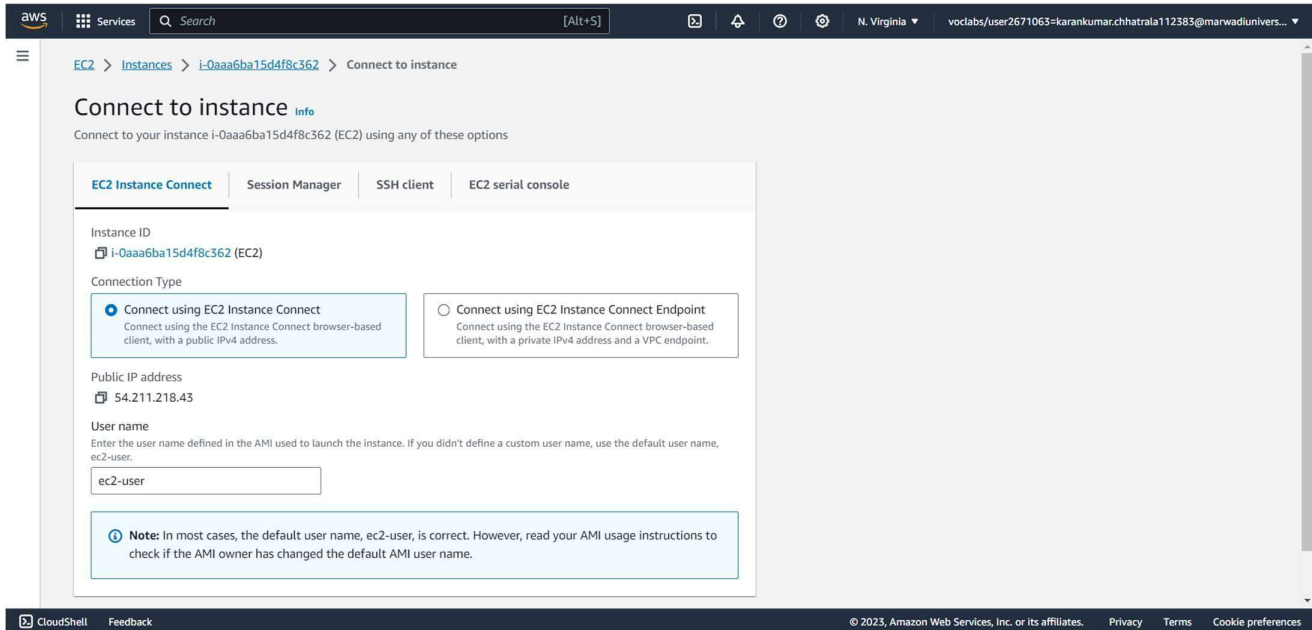
Updated less than a minute ago

Instance ID i-0aaa6ba15d4f8c362 (EC2)	Public IPv4 address 54.211.218.43 open address	Private IPv4 addresses 172.31.28.253
IPv6 address -	Instance state Pending	Public IPv4 DNS ec2-54-211-218-43.compute-1.amazonaws.com open address
Hostname type IP name: ip-172-31-28-253.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-28-253.ec2.internal	Elastic IP addresses -
Answer private resource DNS name IPv4 (A)	Instance type t2.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 54.211.218.43 [Public IP]	VPC ID vpc-0c1e952b1f8823dbe	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0c850e084a93ca6f1	
IMDSv2 Required		

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

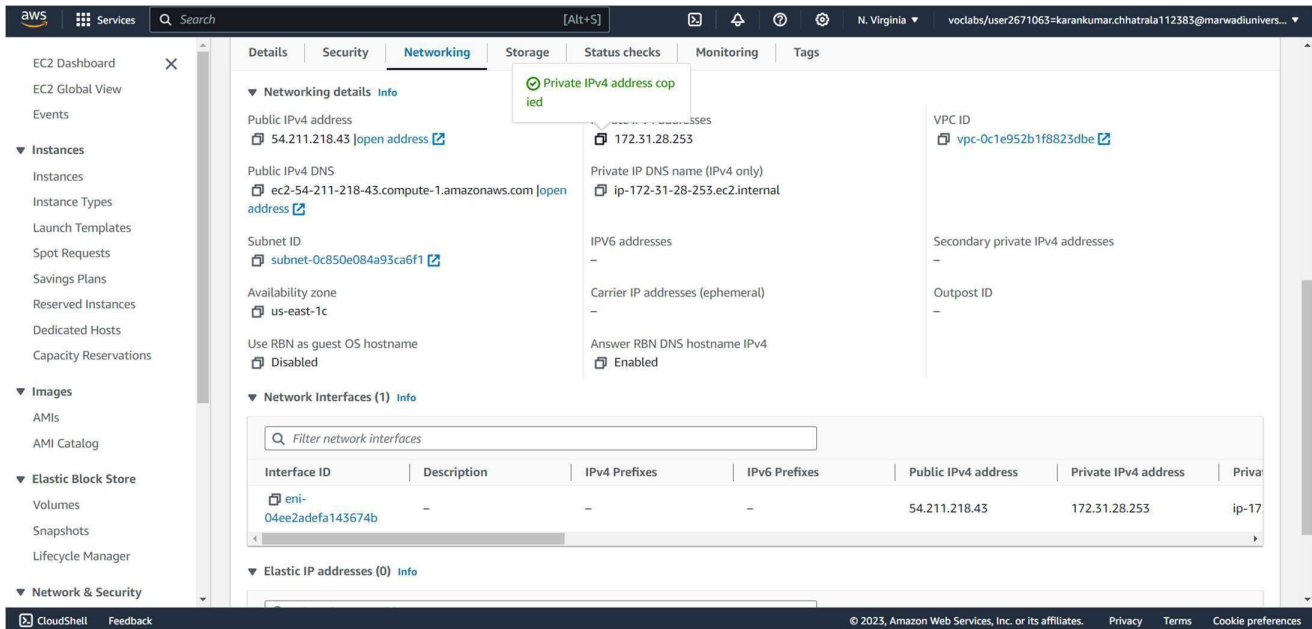
Step 18 : Connect.

Snapshot :



Step 19 : Copy IPv4 address.

Snapshot :





Step 20 : Run ping command Instance 1

Snapshot :

```
[root@ip-172-31-90-87 ec2-user]# ping 44.201.148.121
PING 44.201.148.121 (44.201.148.121) 56(84) bytes of data.
^C
--- 44.201.148.121 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 17682ms

[root@ip-172-31-90-87 ec2-user]# ping 172.31.90.87
PING 172.31.90.87 (172.31.90.87) 56(84) bytes of data.
64 bytes from 172.31.90.87: icmp_seq=1 ttl=127 time=0.020 ms
64 bytes from 172.31.90.87: icmp_seq=2 ttl=127 time=0.036 ms
64 bytes from 172.31.90.87: icmp_seq=3 ttl=127 time=0.036 ms
64 bytes from 172.31.90.87: icmp_seq=4 ttl=127 time=0.034 ms
64 bytes from 172.31.90.87: icmp_seq=5 ttl=127 time=0.035 ms
64 bytes from 172.31.90.87: icmp_seq=6 ttl=127 time=0.039 ms
64 bytes from 172.31.90.87: icmp_seq=7 ttl=127 time=0.033 ms
^C
--- 172.31.90.87 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6215ms
rtt min/avg/max/mdev = 0.020/0.033/0.039/0.005 ms
[root@ip-172-31-90-87 ec2-user]# ^C
[root@ip-172-31-90-87 ec2-user]#
```

Step 21 : Run ping command in Instance 2

Snapshot :

```
[ec2-user@ip-172-31-86-219 ~]$ sudo su
-bash: ~sudo: command not found
[ec2-user@ip-172-31-86-219 ~]$ ping 172.31.90.87
PING 172.31.90.87 (172.31.90.87) 56(84) bytes of data.
```