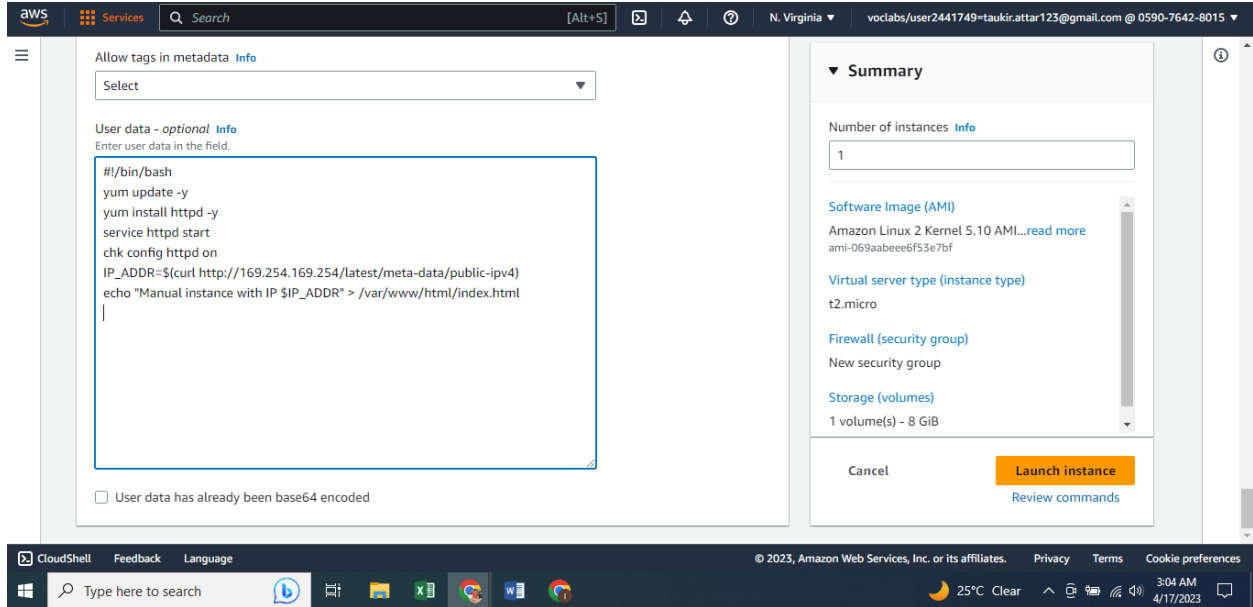
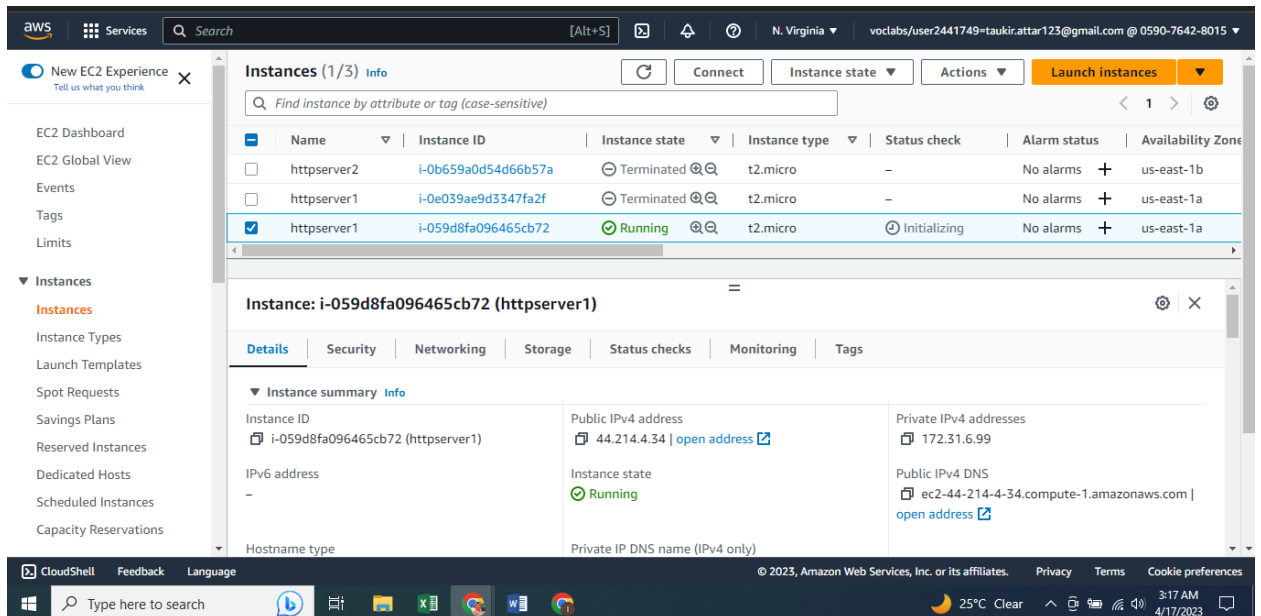


# Project 1 CloudWithAWS-TIO-1-v1.2

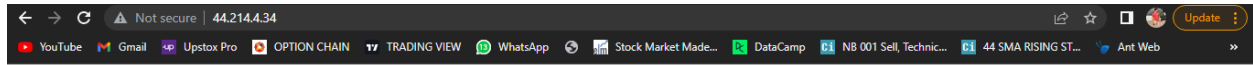
## 1) creating instance



## 2) Instance is created



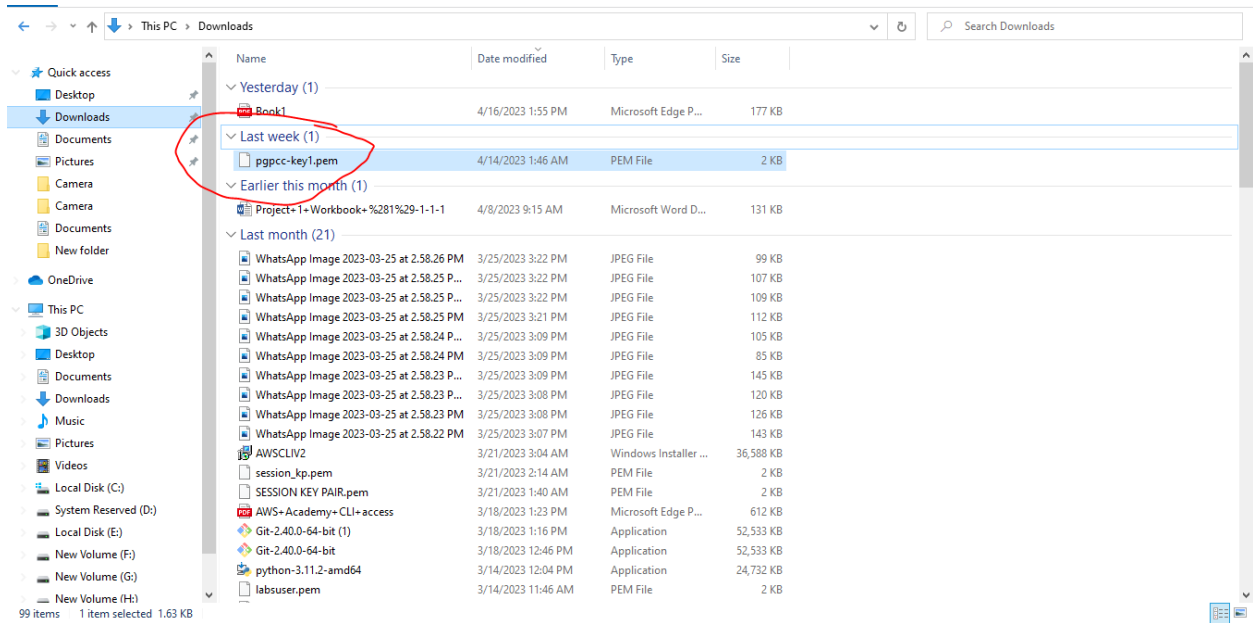
### 3) Public Ip address is pasted in new tab



Manual instance with IP 44.214.4.34

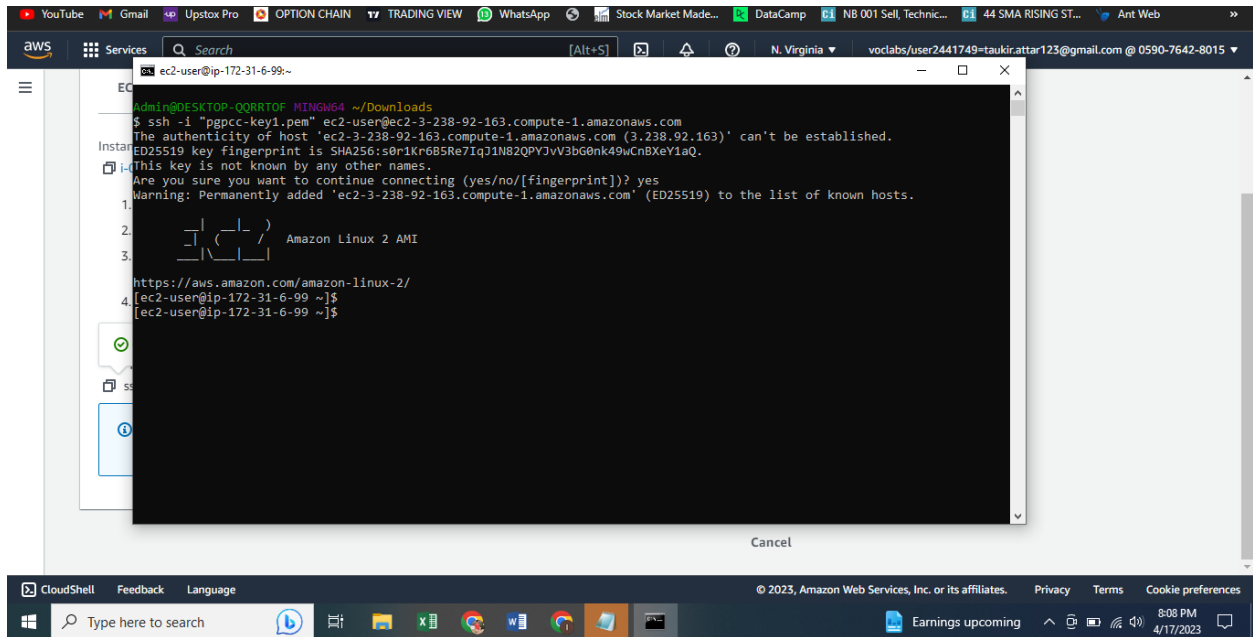


### 4) Pem file location in my system



- 5) `ssh -i "pgpcc-key1.pem" ec2-user@ec2-3-238-92-163.compute-1.amazonaws.com` used this command

got this command from ec2 connect



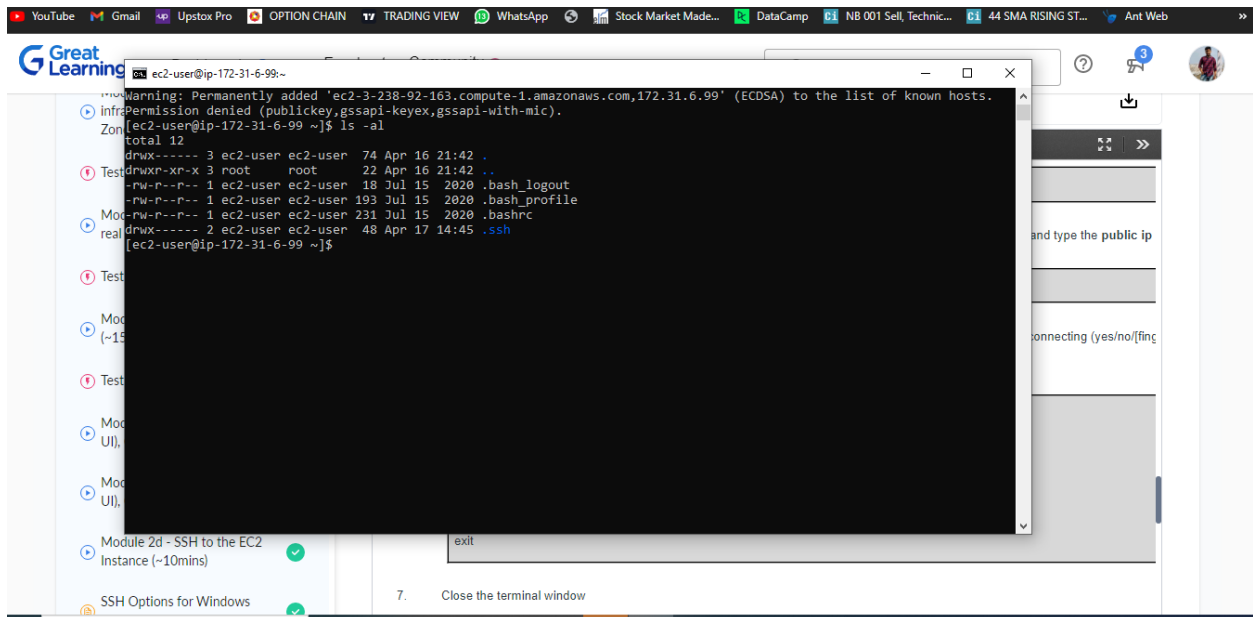
```
Admin@DESKTOP-QORRTOF-MINGW64 ~/Downloads
$ ssh -i "pgpcc-key1.pem" ec2-user@ec2-3-238-92-163.compute-1.amazonaws.com
The authenticity of host 'ec2-3-238-92-163.compute-1.amazonaws.com (3.238.92.163)' can't be established.
ED25519 key fingerprint is SHA256:s0r1Kr6B5Re7IqJ1N82QPv3bG0nk49wCn8XeY1aQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-238-92-163.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

  ____      _
 / ___|    / \
| |  | |  / _ \
| |  | | / ___ \
| |  | |/_/   \_\
|_|  |_____/___)

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-6-99 ~]$
[ec2-user@ip-172-31-6-99 ~]$
```

- `ls -al` run this command



```
Warning: Permanently added 'ec2-3-238-92-163.compute-1.amazonaws.com,172.31.6.99' (ECDSA) to the list of known hosts.
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-172-31-6-99 ~]$ ls -al
total 12
drwxr-xr-x 3 ec2-user ec2-user 74 Apr 16 21:42 .
drwxr-xr-x 3 root root 22 Apr 16 21:42 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
-rw-r--r-- 2 ec2-user ec2-user 48 Apr 17 14:45 .ssh
[ec2-user@ip-172-31-6-99 ~]$
```

- mkdir test
- ls -al
- cd /opt

```

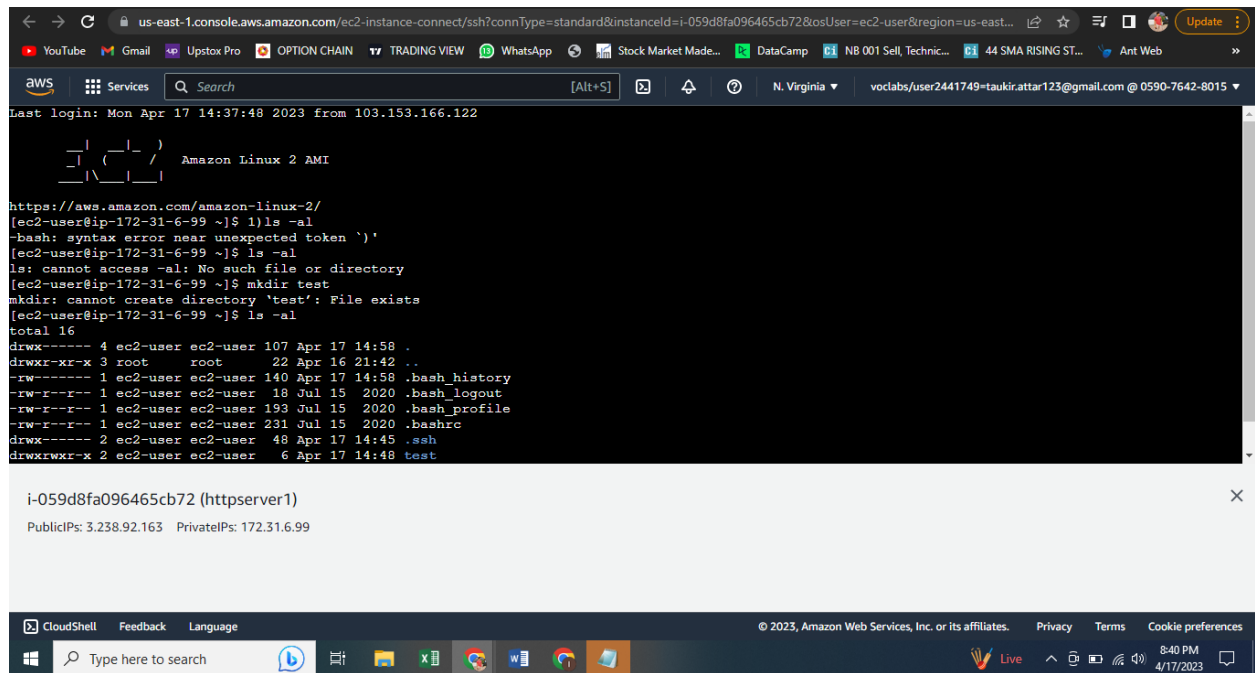
Warning: Permanently added 'ec2-3-238-92-163.compute-1.amazonaws.com,172.31.6.99' (ECDSA) to the list of known hosts.
[ec2-user@ip-172-31-6-99 ~]$ ls -al
total 12
drwxr-xr-x 3 ec2-user ec2-user 74 Apr 16 21:42 .
drwxr-xr-x 3 root root 22 Apr 16 21:42 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
[ec2-user@ip-172-31-6-99 ~]$ mkdir test
[ec2-user@ip-172-31-6-99 ~]$ ls -al
total 12
drwxr-xr-x 4 ec2-user ec2-user 86 Apr 17 14:48 .
drwxr-xr-x 3 root root 22 Apr 16 21:42 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwxr-xr-x 2 ec2-user ec2-user 6 Apr 17 14:48 test
[ec2-user@ip-172-31-6-99 ~]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ exit
logout
Connection to ec2-3-238-92-163.compute-1.amazonaws.com closed.
Admin@DESKTOP-QQRRTOF MINGW64 ~/Downloads
$
  
```

- exit

```

[ec2-user@ip-172-31-6-99 ~]$ mkdir test
[ec2-user@ip-172-31-6-99 ~]$ mkdir test
mkdir: cannot create directory 'test': File exists
[ec2-user@ip-172-31-6-99 ~]$ ls -al
total 12
drwxr-xr-x 4 ec2-user ec2-user 86 Apr 17 14:48 .
drwxr-xr-x 3 root root 22 Apr 16 21:42 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwxr-xr-x 2 ec2-user ec2-user 6 Apr 17 14:48 test
[ec2-user@ip-172-31-6-99 ~]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ cd /opt
[ec2-user@ip-172-31-6-99 opt]$ exit
logout
Connection to ec2-3-238-92-163.compute-1.amazonaws.com closed.
Admin@DESKTOP-QQRRTOF MINGW64 ~/Downloads
$
  
```

## 6) Run the above command



The screenshot shows the AWS CloudShell interface. The terminal window displays the following commands and output:

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-6-99 ~]$ ls -al
-bash: syntax error near unexpected token `)'
[ec2-user@ip-172-31-6-99 ~]$ ls -al
ls: cannot access '-al': No such file or directory
[ec2-user@ip-172-31-6-99 ~]$ mkdir test
mkdir: cannot create directory 'test': File exists
[ec2-user@ip-172-31-6-99 ~]$ ls -al
total 16
drwxr-xr-x 4 ec2-user ec2-user 107 Apr 17 14:58 .
drwxr-xr-x 3 root root 22 Apr 16 21:42 ..
-rw-r--r-- 1 ec2-user ec2-user 140 Apr 17 14:58 .bash_history
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwxr-xr-x 2 ec2-user ec2-user 48 Apr 17 14:45 .ssh
drwxrwxr-x 2 ec2-user ec2-user 6 Apr 17 14:48 test
```

Below the terminal window, the instance details are shown:

i-059d8fa096465cb72 (httpserver1)  
PublicIPs: 3.238.92.163 PrivateIPs: 172.31.6.99

The bottom of the screenshot shows the Windows taskbar with the Start menu, search bar, and various application icons. The system tray shows the date and time as 8:40 PM on 4/17/2023.

## 7) Terminated the instance

## 8) Done with project 1

# Project 2 Cloud With AWS- TIO-2 -v1.1.

## 1) Creating instance 1 , summery of 1<sup>st</sup> instance

The screenshot shows the AWS Management Console interface for creating a new EC2 instance. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information for 'voclabs/user2441749=taukir.attar123@gmail.com' in the 'N. Virginia' region. The main content area is divided into two panels. The left panel, titled 'Allow tags in metadata', has a 'Select' dropdown and a 'User data - optional' section with a text area containing a script: 

```
#!/bin/bash
yumupdate-y
yuminstallhttpd-y
chkconfighttpd
IP_ADDR=$(curlhttp://169.254.169.254/latest/meta-data/publicipv4)
echo"ManualinstancewithIP$IP_ADDR">/var/www/html/index.html
echo"ok">/var/www/html/health.html
```

 Below the text area is a checkbox for 'User data has already been base64 encoded'. The right panel, titled 'Summary', shows the configuration for 1 instance using the 'Amazon Linux 2 AMI (HVM)' image, 't2.micro' instance type, 'New security group' firewall, and '1 volume(s) - 8 GiB' storage. At the bottom of the summary panel are 'Cancel', 'Launch instance', and 'Review commands' buttons.

## 2) Creating 2<sup>nd</sup> instance summary image

The screenshot shows the AWS Management Console interface for creating a second EC2 instance. The top navigation bar is identical to the first screenshot. The main content area is divided into two panels. The left panel shows network and security group configuration. The 'VPC - required' section has a dropdown for 'vpc-0ca2ee4b6488076a7' (default). The 'Subnet' section has a dropdown for 'subnet-0740d4201375f21ca'. The 'Auto-assign public IP' section has a dropdown for 'Enable'. The 'Firewall (security groups)' section has a 'Create security group' button and a 'Select existing security group' radio button. The 'Security group name - required' section has a text input for 'launch-wizard-6'. The 'Description - required' section has a text input for 'launch-wizard-6 created 2023-04-16T09:46:28.042Z'. The right panel, titled 'Summary', shows the configuration for 1 instance using the 'Amazon Linux 2 AMI (HVM)' image, 't2.micro' instance type, 'New security group' firewall, and '1 volume(s) - 8 GiB' storage. At the bottom of the summary panel are 'Cancel', 'Launch instance', and 'Review commands' buttons.

### 3) State of both the instances are running

The screenshot displays the AWS Management Console interface for EC2 instances. The left sidebar shows navigation options like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, and Instances. The main content area shows a list of instances under the heading 'Instances (1/2) Info'. Two instances are listed: 'httpserver2' and 'httpserver1'. Both are in a 'Running' state with '2/2 checks passed'. Below the list, the details for 'httpserver1' (Instance ID: i-0e039ae9d3347fa2f) are expanded, showing its public IPv4 address (3.235.9.93), private IPv4 address (172.31.12.216), and public IPv4 DNS (ec2-3-235-9-93.compute-1.amazonaws.com).

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
httpserver2	i-0b659a0d54d66b57a	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b
httpserver1	i-0e039ae9d3347fa2f	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a

**Instance: i-0e039ae9d3347fa2f (httpserver1)**

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
<b>Instance summary Info</b>						
Instance ID	i-0e039ae9d3347fa2f (httpserver1)		Public IPv4 address	3.235.9.93   <a href="#">open address</a>		
IPv6 address	-		Instance state	Running		
Hostname type	Private IP DNS name (IPv4 only)		Private IPv4 addresses	172.31.12.216		
			Public IPv4 DNS	ec2-3-235-9-93.compute-1.amazonaws.com   <a href="#">open address</a>		

### 4) Creating target group

The screenshot displays the AWS Management Console interface for creating a target group. The left sidebar shows navigation options like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, and Instances. The main content area shows the 'Target groups (1/1) Info' page. A table lists the existing target group 'web-tg' with an ARN of 'arn:aws:elasticloadbalancing...', port 80, HTTP protocol, and Instance target type. Below the table, the details for 'web-tg' are expanded, showing the 'Targets' tab with a list of instances.

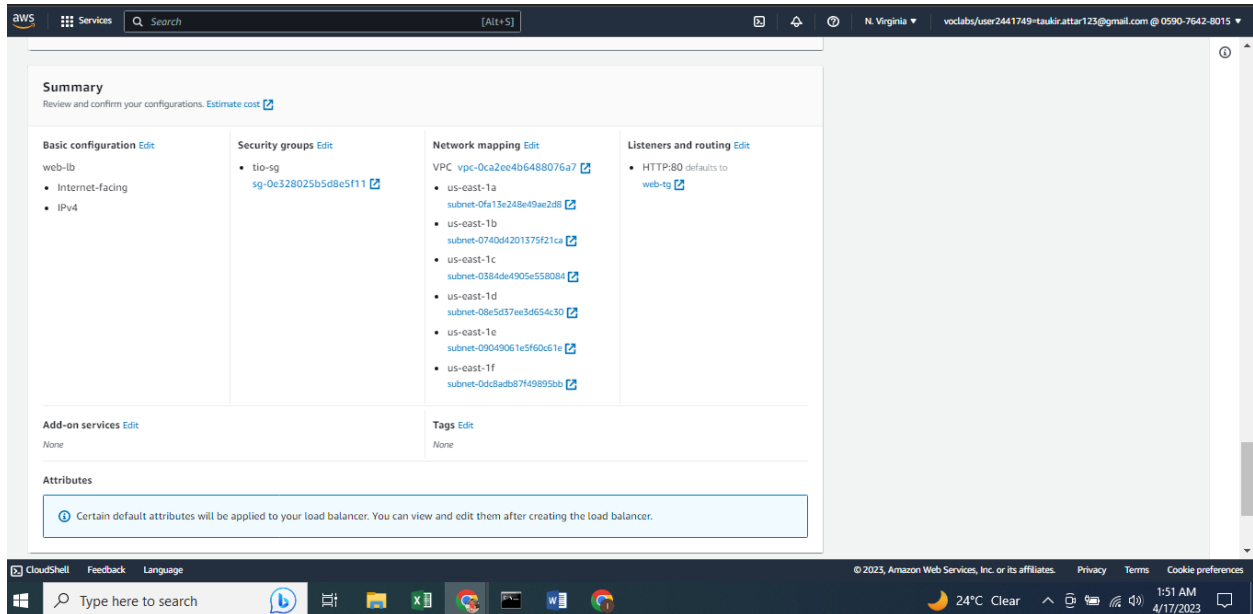
Name	ARN	Port	Protocol	Target type	Load balancer
web-tg	arn:aws:elasticloadbalancing...	80	HTTP	Instance	None associated

**Target group: web-tg**

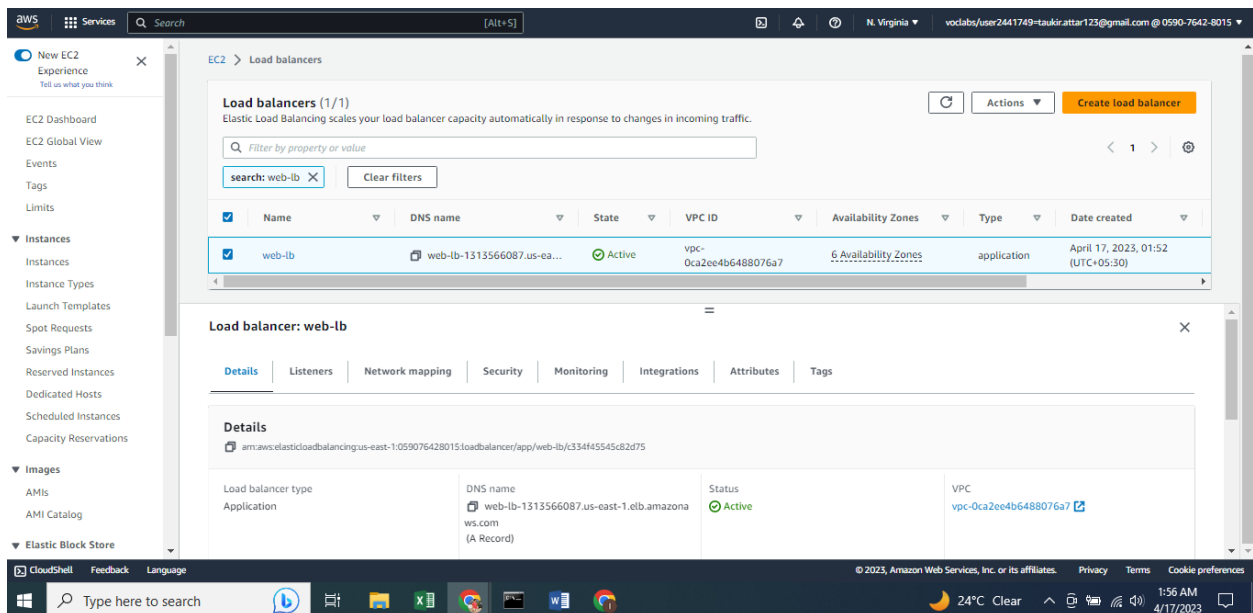
Details	Targets	Monitoring	Health checks	Attributes	Tags
<b>Details</b>					

## 5) Creating load balancer .

### Summary of load balancer imag



## 6) LOAD BALANCER IS CREATED



## 7) This is the dns

<http://web-lb-1313566087.us-east-1.elb.amazonaws.com/>



- 8) Deleted the target group
- 9) Deleted the load balancer
- 10) Terminated both the instances
- 11) Project 2 is completed