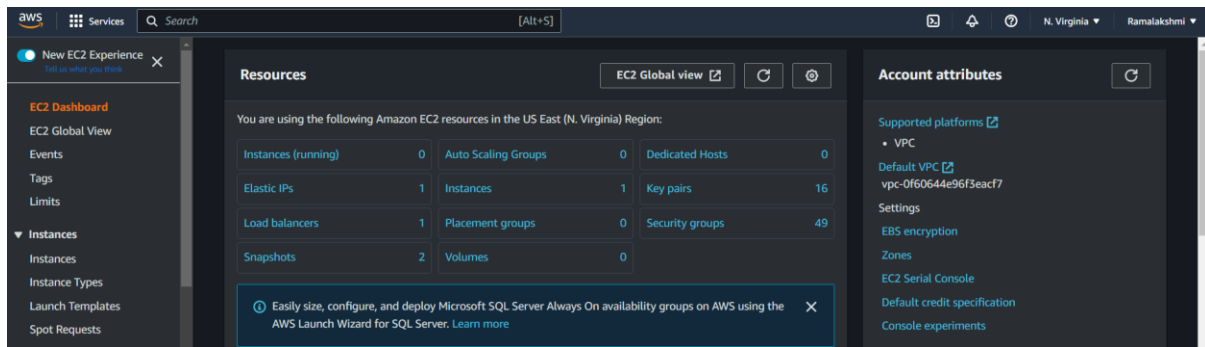
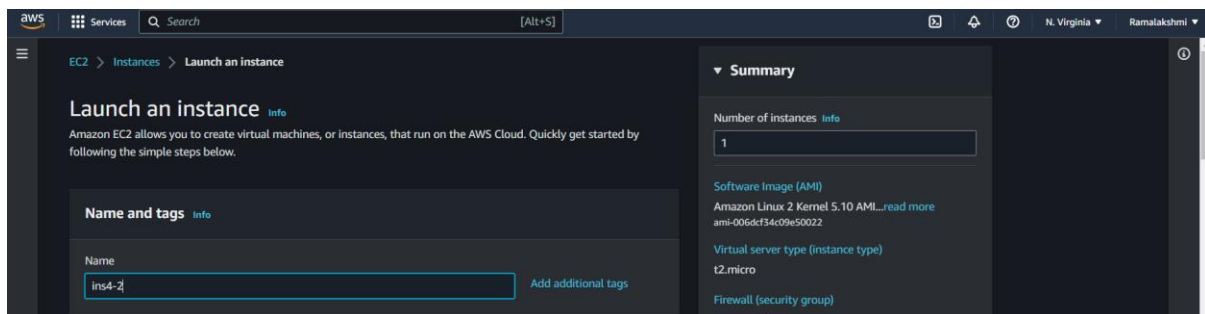


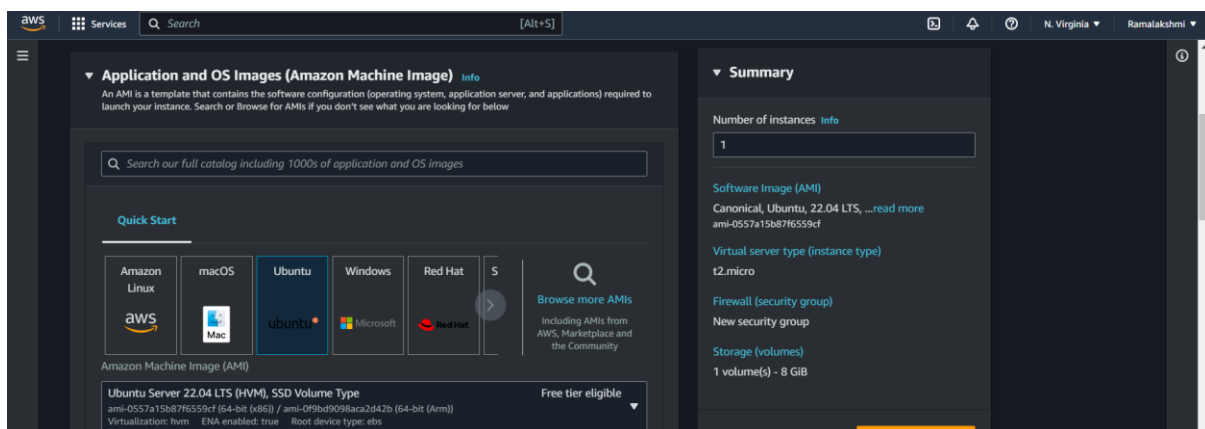
## Step 1: Create an EC2 instance using ubuntu AMI.



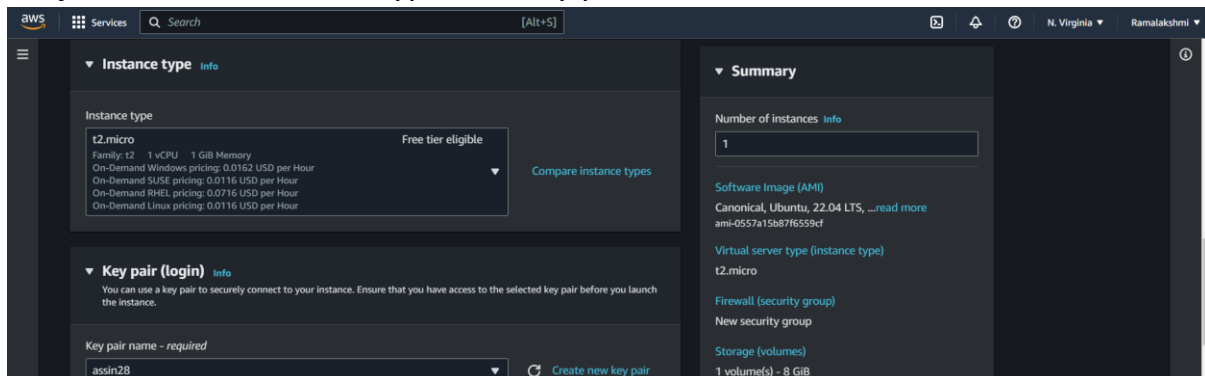
### Giving ec2 instance name



### Choosing AMI



## Step 2: Choose instance type and key pair.



### Step 3: Creating a security group.

Network [Info](#)  
vpc-0f60644e96f3eac7

Subnet [Info](#)  
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)  
Enable

Firewall (security groups) [Info](#)  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-22' with the following rules:

- ☒ Allow SSH traffic from [Info](#)  
Helps you connect to your instance. [Anywhere](#) [0.0.0.0/0](#)
- ☒ Allow HTTPS traffic from the internet  
To set up an endpoint, for example when creating a web server
- ☒ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

[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.](#)

**Summary**

Number of instances [Info](#)  
1

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-0557a15b87f6559cf

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#)

### Step 4: Adding the User data in additional settings and launch instance.

Metadata response hop limit [Info](#)  
[Select](#)

Allow tags in metadata [Info](#)  
[Select](#)

User data - optional [Info](#)  
Enter user data in the field.

```
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
IP_ADDR=$(curl http://169.254.169.254/latest/meta-data/public-ipv4)
echo "Manual instance with IP $IP_ADDR" > /var/www/html/index.html
```

☐ User data has already been base64 encoded

**Summary**

Number of instances [Info](#)  
1

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-0557a15b87f6559cf

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#)

### Instance got created

**Success**  
Successfully initiated launch of instance (i-07d0b23c1bf057882)

[Launch log](#)

**Next Steps**

Create billing and free tier usage alerts  
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.  
[Create billing alerts](#)

Connect to your instance  
Once your instance is running, log into it from your local computer.  
[Connect to instance](#)  
[Learn more](#)

Connect an RDS database  
Configure the connection between an EC2 instance and a database to allow traffic flow between them.  
[Connect an RDS database](#)  
[Create a new RDS database](#)  
[Learn more](#)

## Step 5: Now check the status of the instance.

The screenshot shows the AWS Management Console interface. On the left, the navigation menu includes 'New EC2 Experience', 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Scheduled Instances', 'Capacity Reservations', 'Images', and 'AMIs'. The main content area is titled 'Instances (1/1) Info'. It features a search bar with the text 'Find instance by attribute or tag (case-sensitive)' and a filter button. Below the search bar, a table lists instances. The first instance, 'ins4-2', is highlighted. Its details are shown in a modal window titled 'Instance: i-07d0b23c1bf057882 (ins4-2)'. The modal has tabs for 'Details', 'Security', 'Networking', 'Storage', 'Status checks', 'Monitoring', and 'Tags'. The 'Details' tab is active, showing the 'Instance summary' with the following information:

Instance summary		
Instance ID	Public IPv4 address	Private IPv4 addresses
i-07d0b23c1bf057882 (ins4-2)	18.204.18.77   <a href="#">open address</a>	172.31.86.239
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-18-204-18-77.compute-1.amazonaws.com   <a href="#">open address</a>
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-86-239.ec2.internal	ip-172-31-86-239.ec2.internal	
Answer private resource DNS name	Instance type	

The screenshot shows the 'Connect to instance' page in the AWS Management Console. The page title is 'Connect to instance info'. Below the title, it says 'Connect to your instance i-07d0b23c1bf057882 (ins4-2) using any of these options'. There are four tabs: 'EC2 Instance Connect', 'Session Manager', 'SSH client', and 'EC2 serial console'. The 'SSH client' tab is selected. It displays the instance ID 'i-07d0b23c1bf057882 (ins4-2)' and a list of steps to connect via SSH:

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is `assin28.pem`.
3. Run this command, if necessary, to ensure your key is not publicly viewable.  
`chmod 400 assin28.pem`
4. Connect to your instance using its Public DNS:  
`ec2-18-204-18-77.compute-1.amazonaws.com`

Below the steps, there is an 'Example:' section with the command:

```
ssh -i "assin28.pem" ubuntu@ec2-18-204-18-77.compute-1.amazonaws.com
```

A note at the bottom states: 'Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.'

**Step 7:** To connect SSH client locally Download Git Bash and go to the git bash and open git bash cli and paste the command copied in the ssh client command. Here we see that EC2 instance is connected with git bash successfully.

```
RAMALAKSHMI@DESKTOP-ADOQDL7 MINGW64 ~ (branchone)
$ cd downloads

RAMALAKSHMI@DESKTOP-ADOQDL7 MINGW64 ~/downloads (branchone)
$ ssh -i "assin28.pem" ubuntu@ec2-18-204-18-77.compute-1.amazonaws.com
The authenticity of host 'ec2-18-204-18-77.compute-1.amazonaws.com (18.204.18.77)' can't be established.
ED25519 key fingerprint is SHA256:gRvXukARqE1mQIL7LqD1nhnNcB/sEHj0zPilsY9015M.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-204-18-77.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1028-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Sat Mar  4 16:29:38 UTC 2023

System load:  0.0               Processes:           98
Usage of /:   20.0% of 7.57GB   Users logged in:    0
Memory usage: 19%              IPv4 address for eth0: 172.31.86.239
Swap usage:   0%

 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-86-239:~$
```

**Step 8:** To install nodejs we to update apt package manager using the command `sudo apt update`.

```
ubuntu@ip-172-31-86-239:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [939 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [203 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [13.6 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [680 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [106 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [584 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [877 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [172 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [17.9 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [9696 B]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [3260 B]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [444 B]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [40.7 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [9800 B]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [392 B]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [19.5 kB]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [14.0 kB]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [392 B]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [680 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [139 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [8528 B]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [637 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [99.7 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [696 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [111 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [13.5 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [4960 B]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [996 B]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [240 B]
Fetched 26.2 MB in 5s (5785 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
38 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-86-239:~$
```

**Step 9:** Now install the Nodejs using the command `sudo apt install nodejs..`

```
ubuntu@ip-172-31-86-239:~$ sudo apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  javascript-common libc-ares2 libjs-highlight.js libnode72 nodejs-doc
Suggested packages:
  apache2 | lighttpd | httpd npm
The following NEW packages will be installed:
  javascript-common libc-ares2 libjs-highlight.js libnode72 nodejs nodejs-doc
0 upgraded, 6 newly installed, 0 to remove and 38 not upgraded.
Need to get 13.7 MB of archives.
After this operation, 53.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 javascript-common all 11+nmu1 [5936 B]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 libjs-highlight.js all 9.18.5+dfsg1-1 [367 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc-ares2 amd64 1.18.1-1ubuntu0.22.04.1 [45.1 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 libnode72 amd64 12.22.9~dfsg-1ubuntu3 [10.8 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 nodejs-doc all 12.22.9~dfsg-1ubuntu3 [2409 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 nodejs amd64 12.22.9~dfsg-1ubuntu3 [122 kB]
Fetched 13.7 MB in 0s (43.7 MB/s)
Selecting previously unselected package javascript-common.
(Reading database ... 63605 files and directories currently installed.)
Preparing to unpack .../0-javascript-common_11+nmu1_all.deb ...
Unpacking javascript-common (11+nmu1) ...
Selecting previously unselected package libjs-highlight.js.
Preparing to unpack .../1-libjs-highlight.js_9.18.5+dfsg1-1_all.deb ...
Unpacking libjs-highlight.js (9.18.5+dfsg1-1) ...
Selecting previously unselected package libc-ares2:amd64.
Preparing to unpack .../2-libc-ares2_1.18.1-1ubuntu0.22.04.1_amd64.deb ...
Unpacking libc-ares2:amd64 (1.18.1-1ubuntu0.22.04.1) ...
Selecting previously unselected package libnode72:amd64.
Preparing to unpack .../3-libnode72_12.22.9~dfsg-1ubuntu3_amd64.deb ...
Unpacking libnode72:amd64 (12.22.9~dfsg-1ubuntu3) ...
Selecting previously unselected package nodejs-doc.
Preparing to unpack .../4-nodejs-doc_12.22.9~dfsg-1ubuntu3_all.deb ...
Unpacking nodejs-doc (12.22.9~dfsg-1ubuntu3) ...
Selecting previously unselected package nodejs.
Preparing to unpack .../5-nodejs_12.22.9~dfsg-1ubuntu3_amd64.deb ...
Unpacking nodejs (12.22.9~dfsg-1ubuntu3) ...
Setting up javascript-common (11+nmu1) ...
Setting up libc-ares2:amd64 (1.18.1-1ubuntu0.22.04.1) ...
Setting up libnode72:amd64 (12.22.9~dfsg-1ubuntu3) ...
Setting up libjs-highlight.js (9.18.5+dfsg1-1) ...
Setting up nodejs (12.22.9~dfsg-1ubuntu3) ...
update-alternatives: using /usr/bin/nodejs to provide /usr/bin/js (js) in auto mode
Setting up nodejs-doc (12.22.9~dfsg-1ubuntu3) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

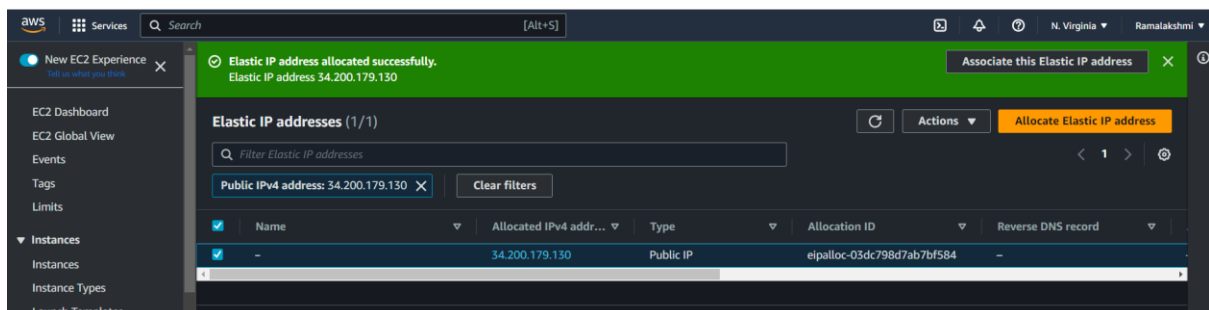
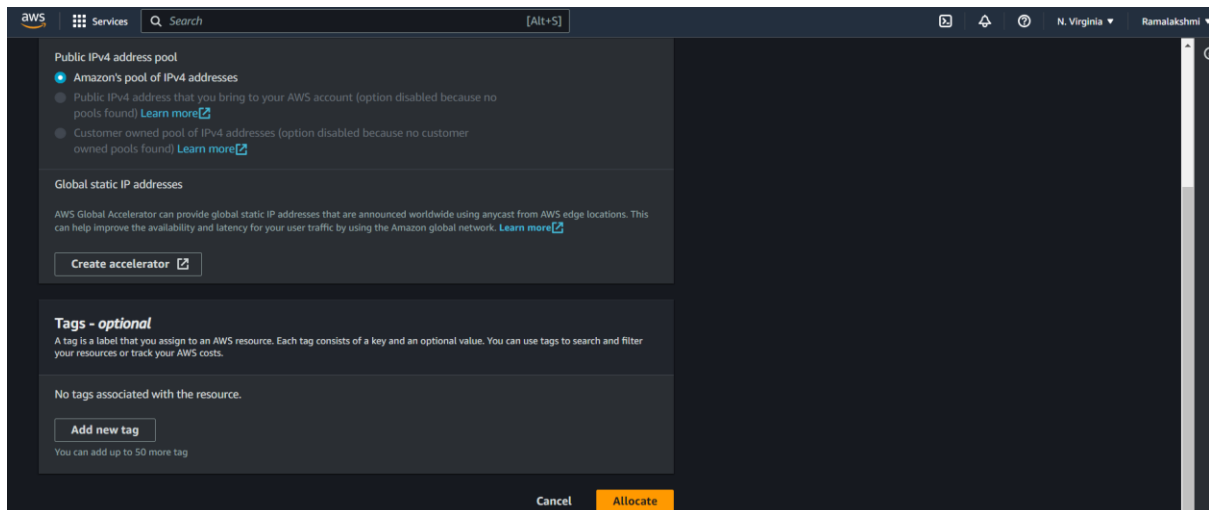
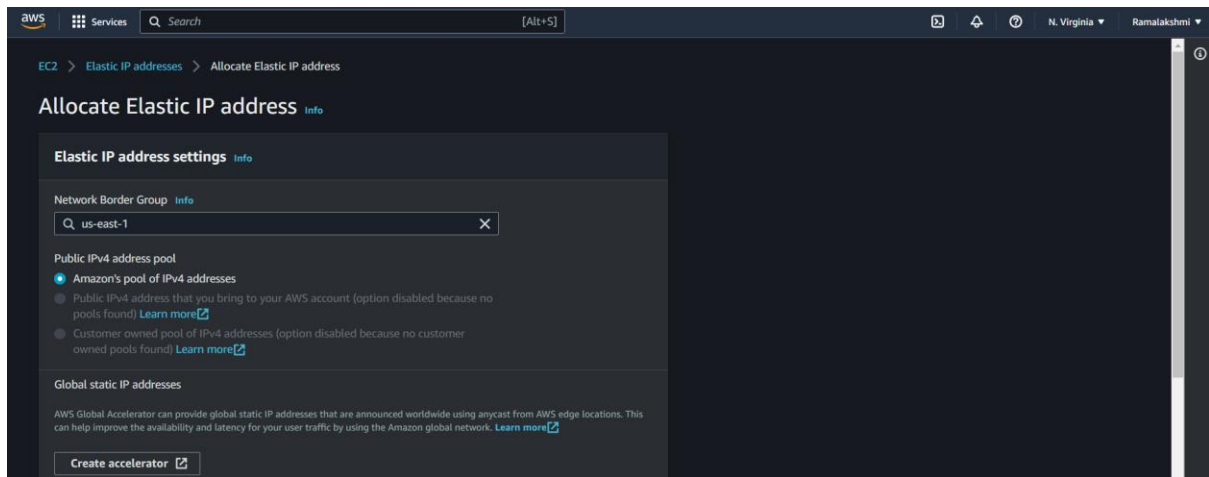
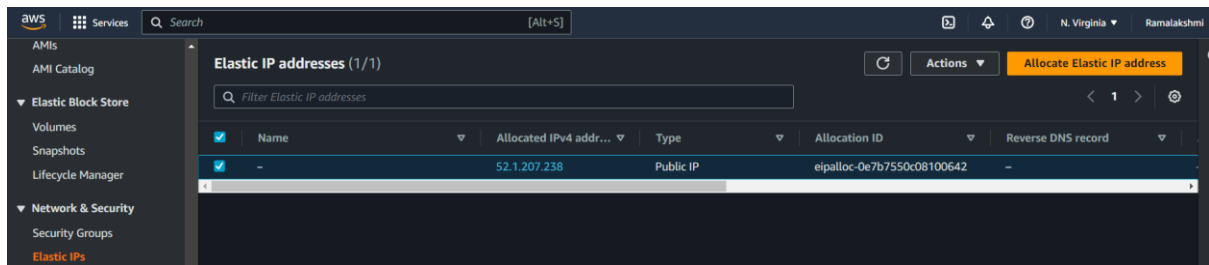
No services need to be restarted.

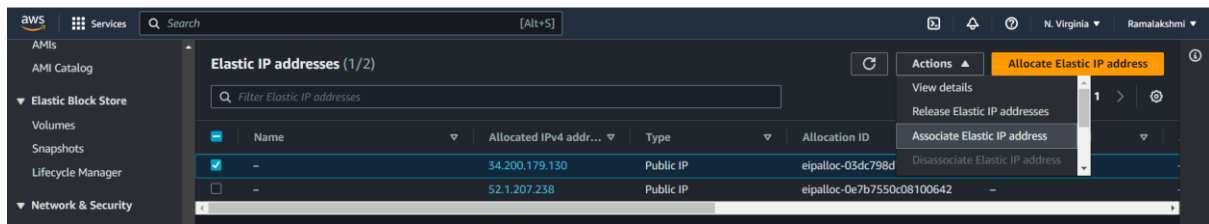
No containers need to be restarted.
```

**Step 11:** After installing to check the version of the nodejs using the command `node -v`.

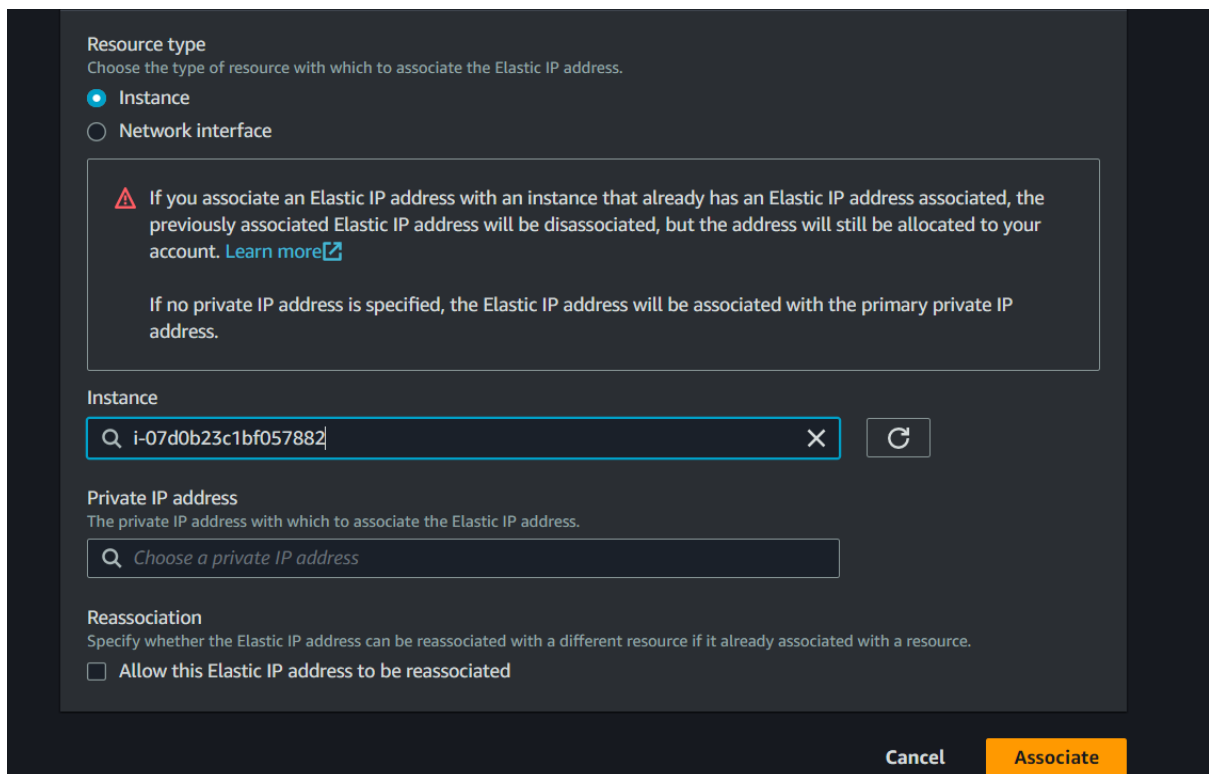
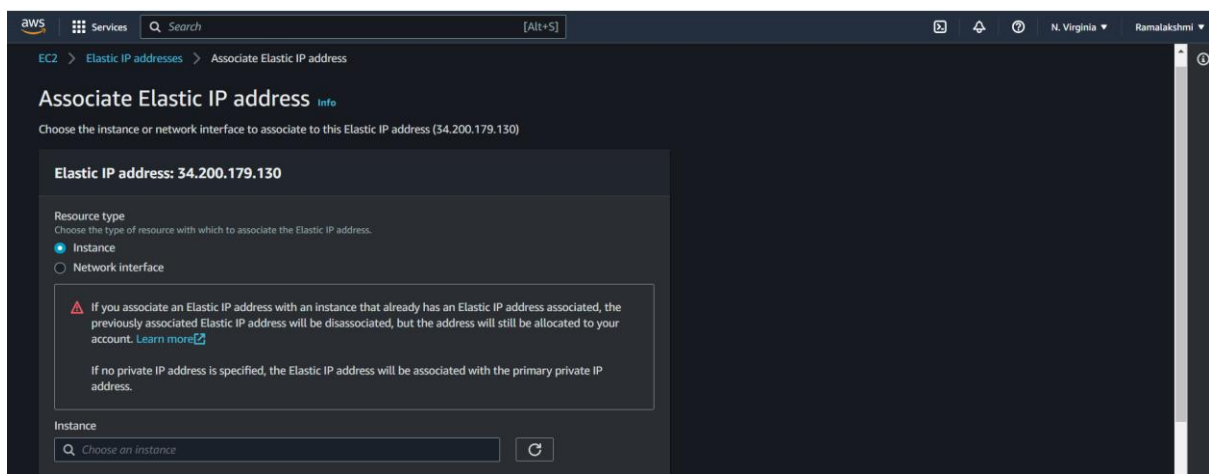
```
ubuntu@ip-172-31-86-239:~$ node -v
v12.22.9
ubuntu@ip-172-31-86-239:~$
```

**Step12:** Now allocate Elastic IP address for instance.

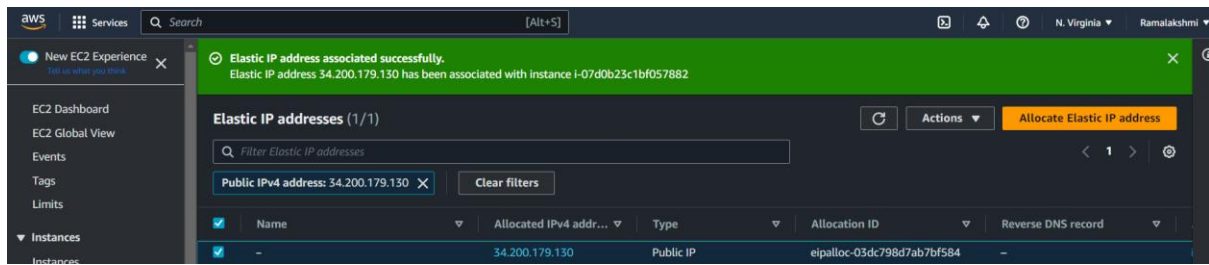




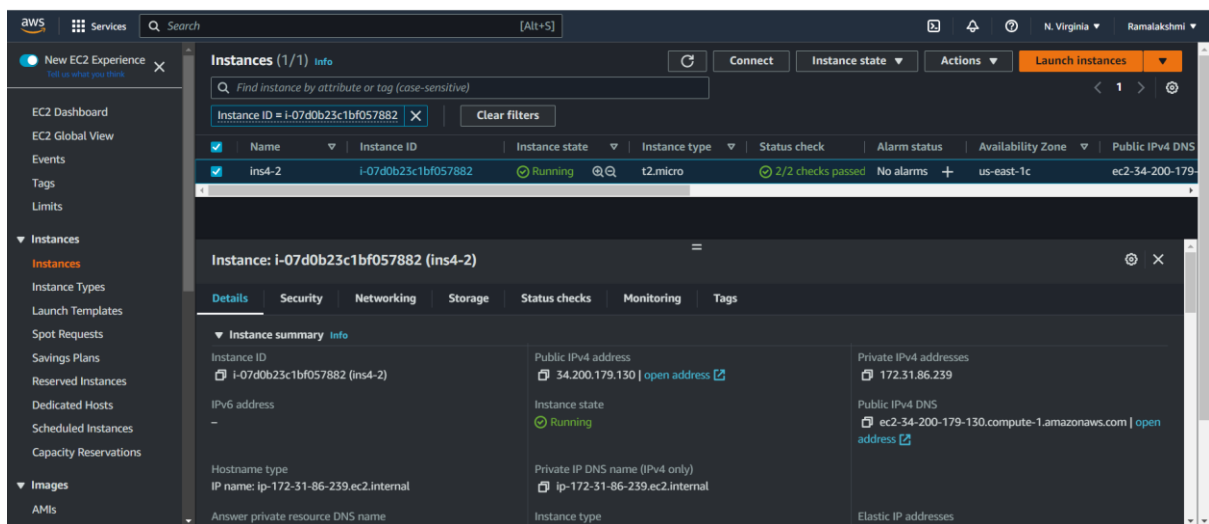
**Step 13:** Now associate with the created instance.



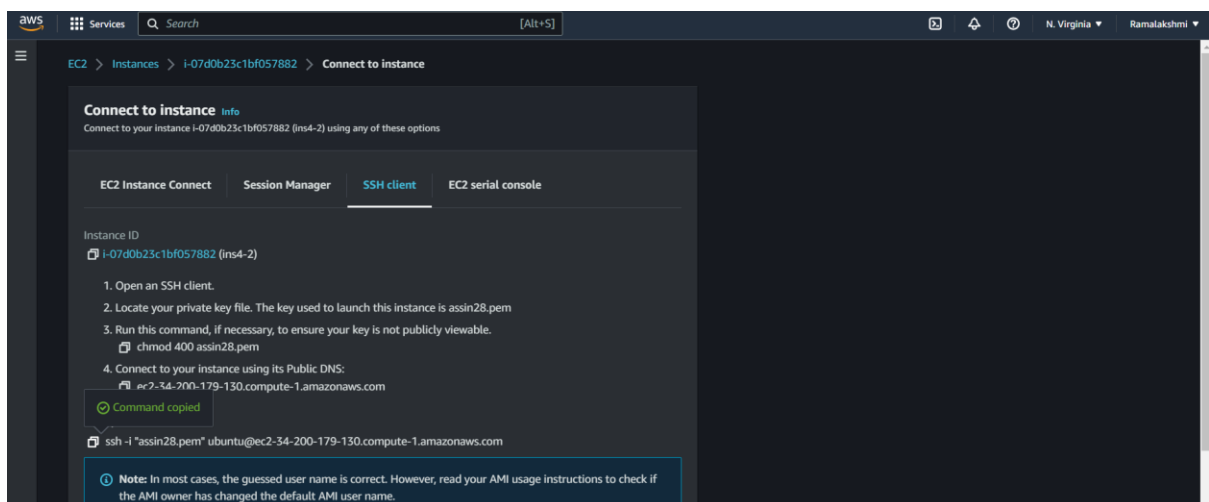




**Step 14:** Here we check that Elastic IP is created.



**Step 15:** Now connect an instance with the Git Bash



**Step 16:** Here we check that Elastic Ip address instance and configuring the IP address.

```
RAMALAKSHMI@DESKTOP-ADOQDL7 MINGW64 ~/downloads (branchone)
$ chmod 400 assin28.pem

RAMALAKSHMI@DESKTOP-ADOQDL7 MINGW64 ~/downloads (branchone)
$ ssh -i "assin28.pem" ubuntu@ec2-34-200-179-130.compute-1.amazonaws.com
The authenticity of host 'ec2-34-200-179-130.compute-1.amazonaws.com (34.200.179.130)' can't be established.
ED25519 key fingerprint is SHA256:gRvXukARqElmQil7LqDlnhnNcB/SEHjozPilsY9015M.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:8: ec2-18-204-18-77.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-34-200-179-130.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1028-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Sat Mar  4 16:46:10 UTC 2023

System load:  0.0               Processes:           98
Usage of /:   23.1% of 7.57GB   Users logged in:    1
Memory usage: 23%              IPv4 address for eth0: 172.31.86.239
Swap usage:   0%

 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

   https://ubuntu.com/aws/pro

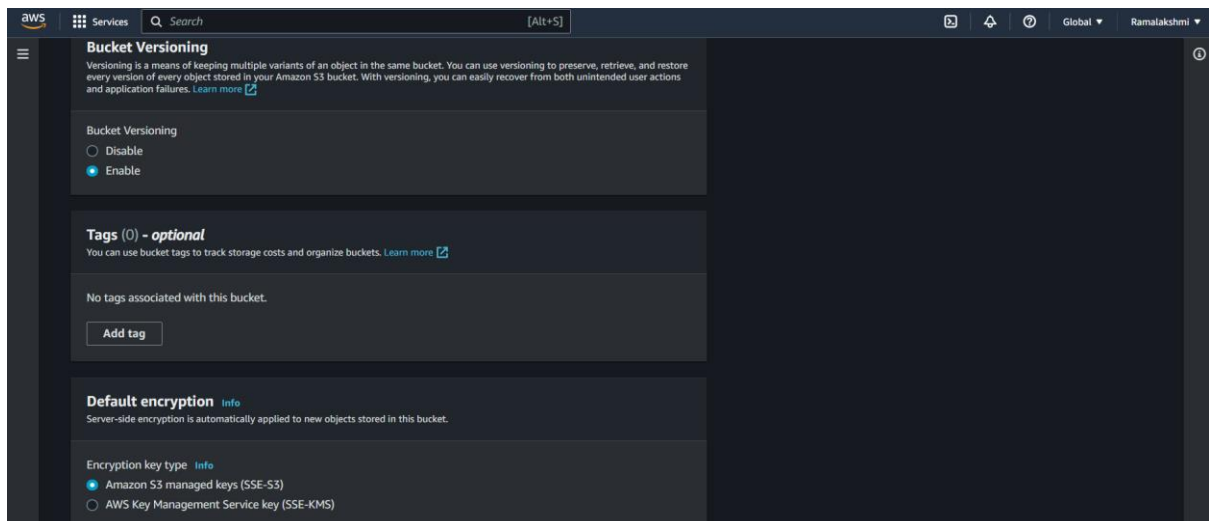
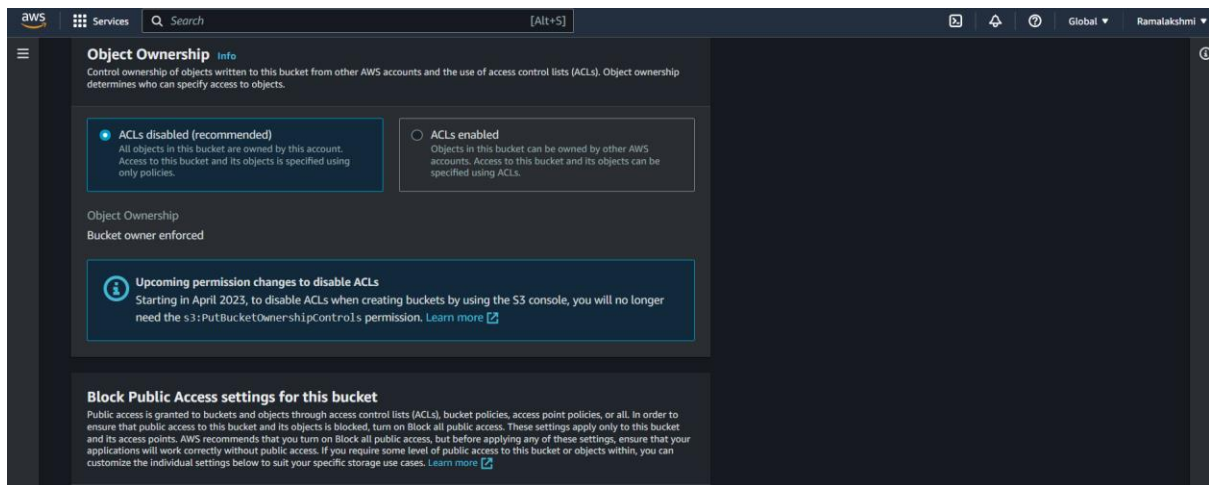
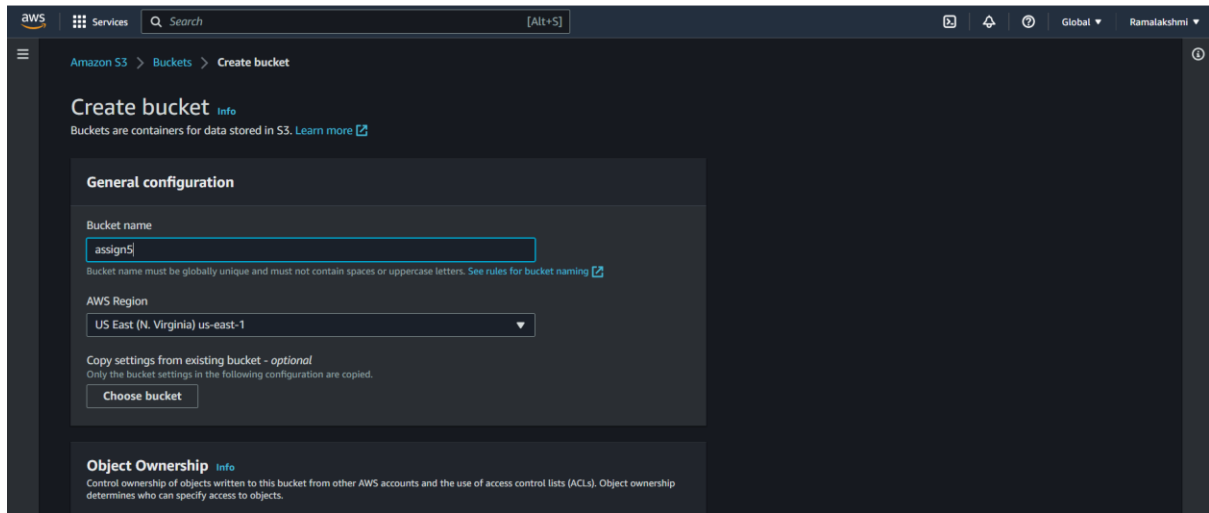
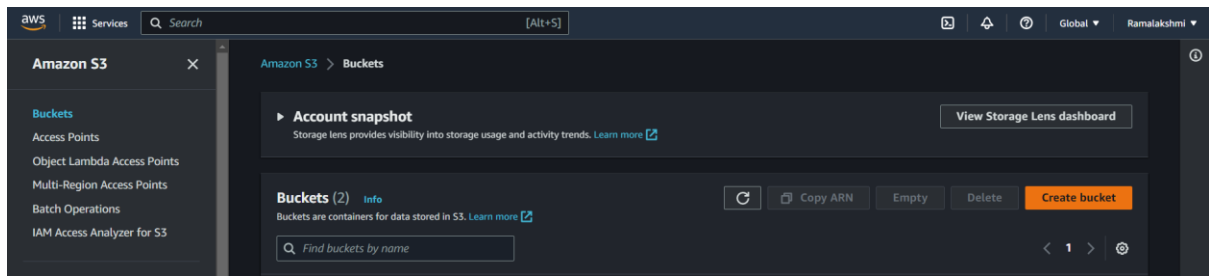
Expanded Security Maintenance for Applications is not enabled.

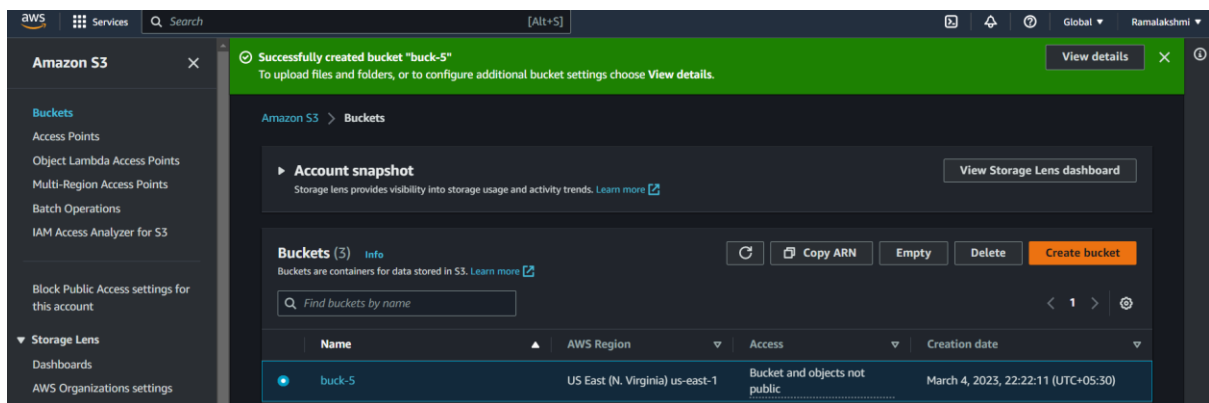
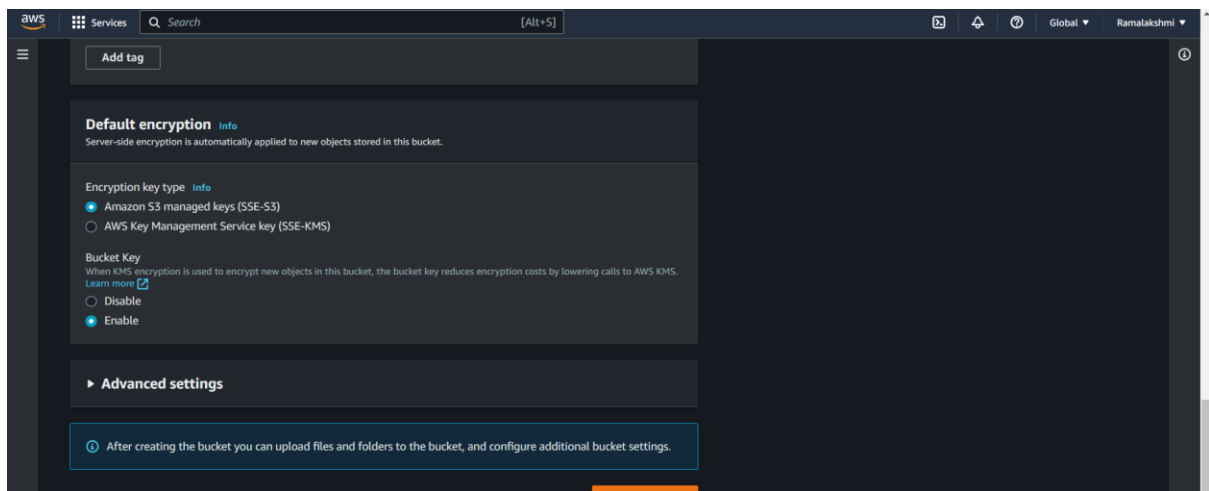
37 updates can be applied immediately.
18 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

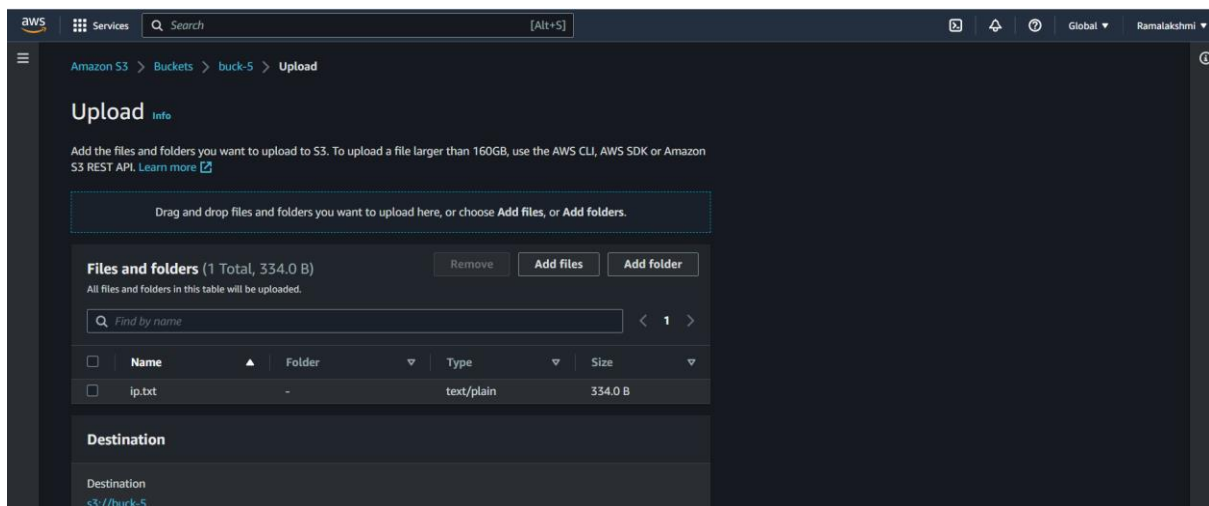
Last login: Sat Mar  4 16:29:39 2023 from 157.47.79.190
ubuntu@ip-172-31-86-239:~$ |
```

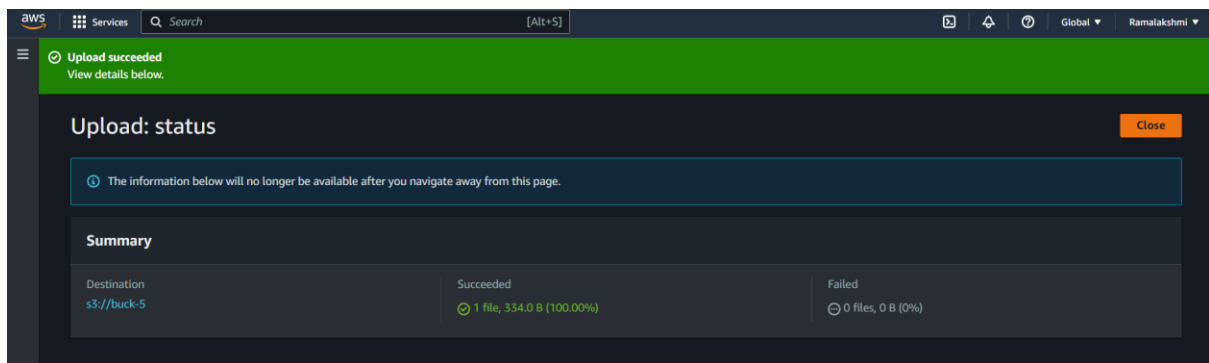
**Step 17:** Now we are creating S3 bucket.





**Step 18:** Now upload a file in the bucket.





**Step 19:** Now copy the URL of the object and open in the browser.

<https://buck-5.s3.amazonaws.com/ip.txt>

