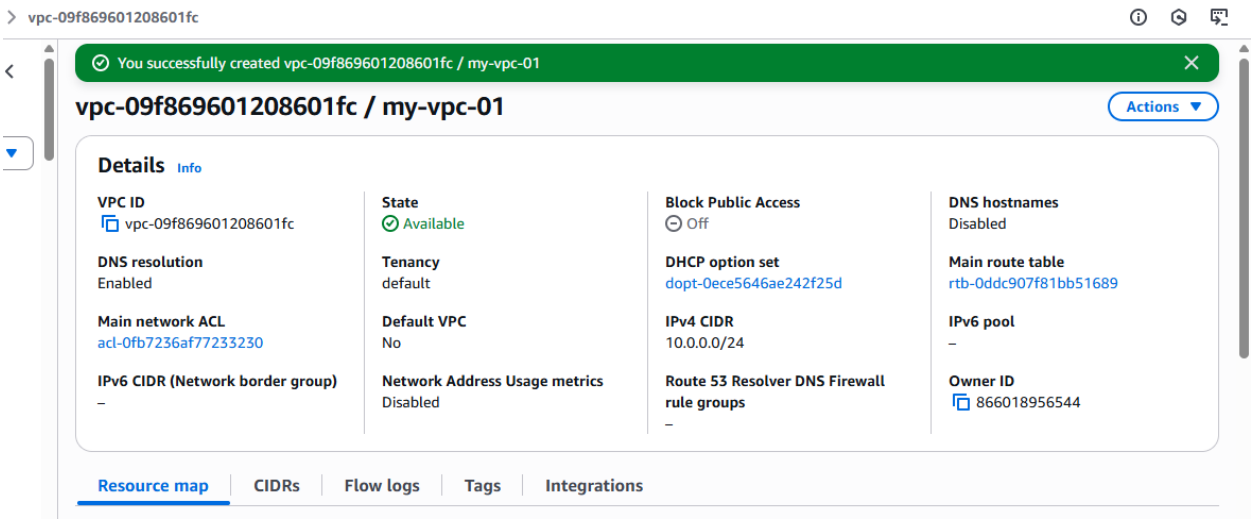
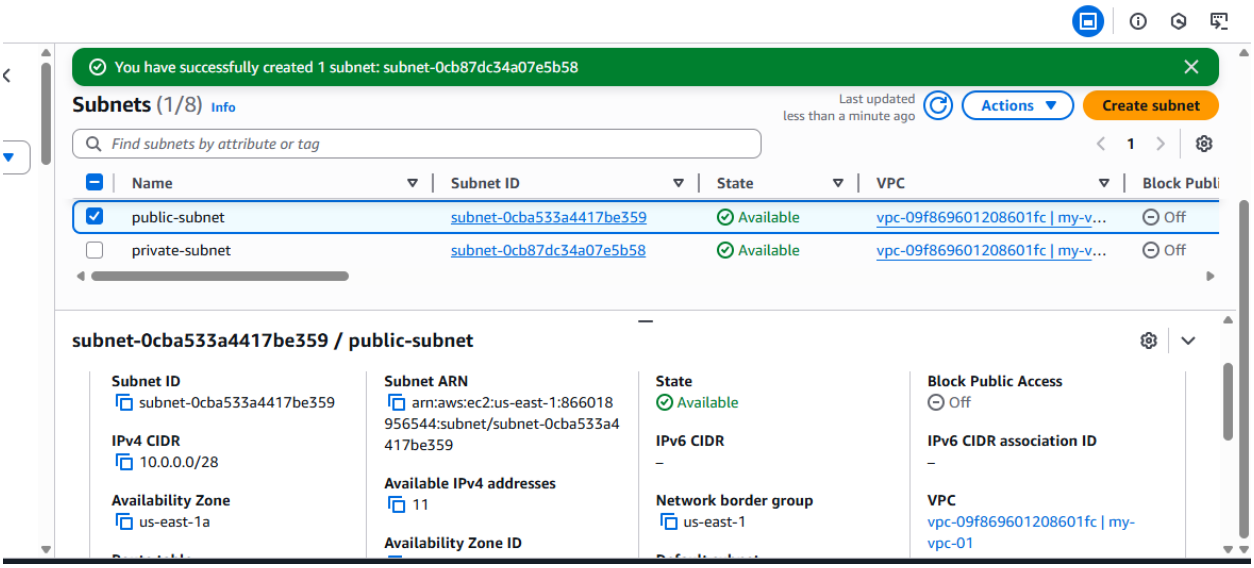
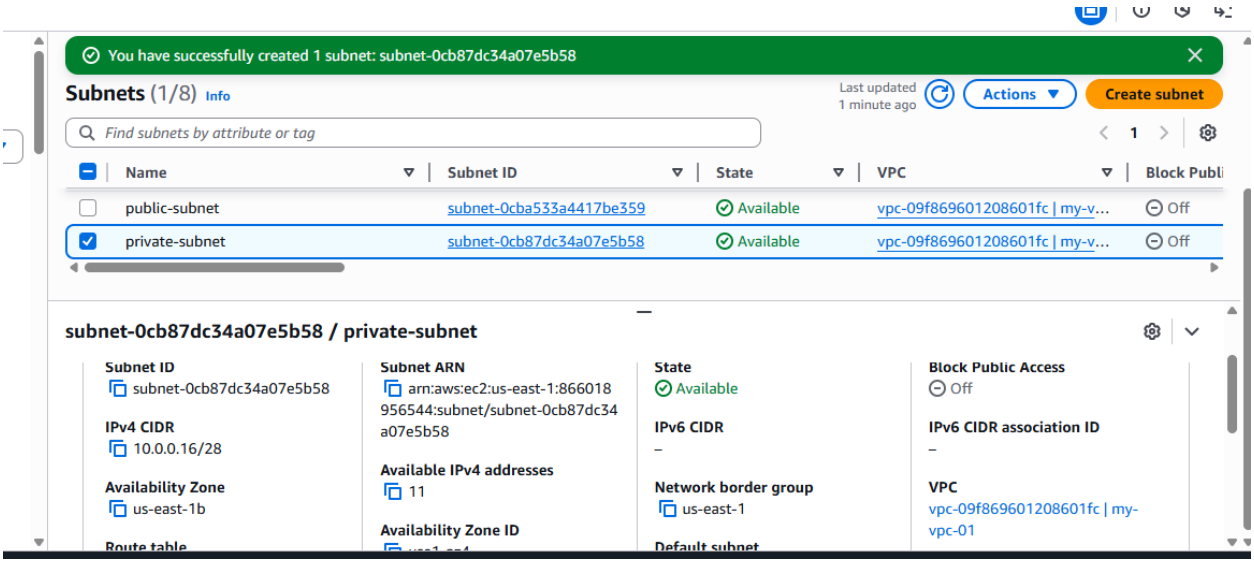
1) Create one Vpc in N.Virginia region.

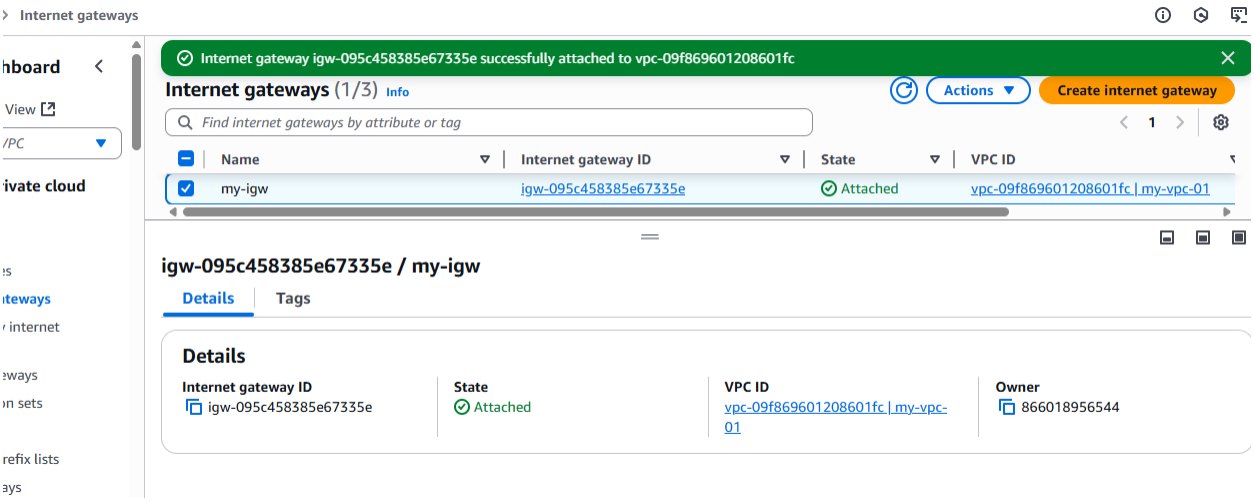


2) Create two subnets. One Public subnet and one private subnet.

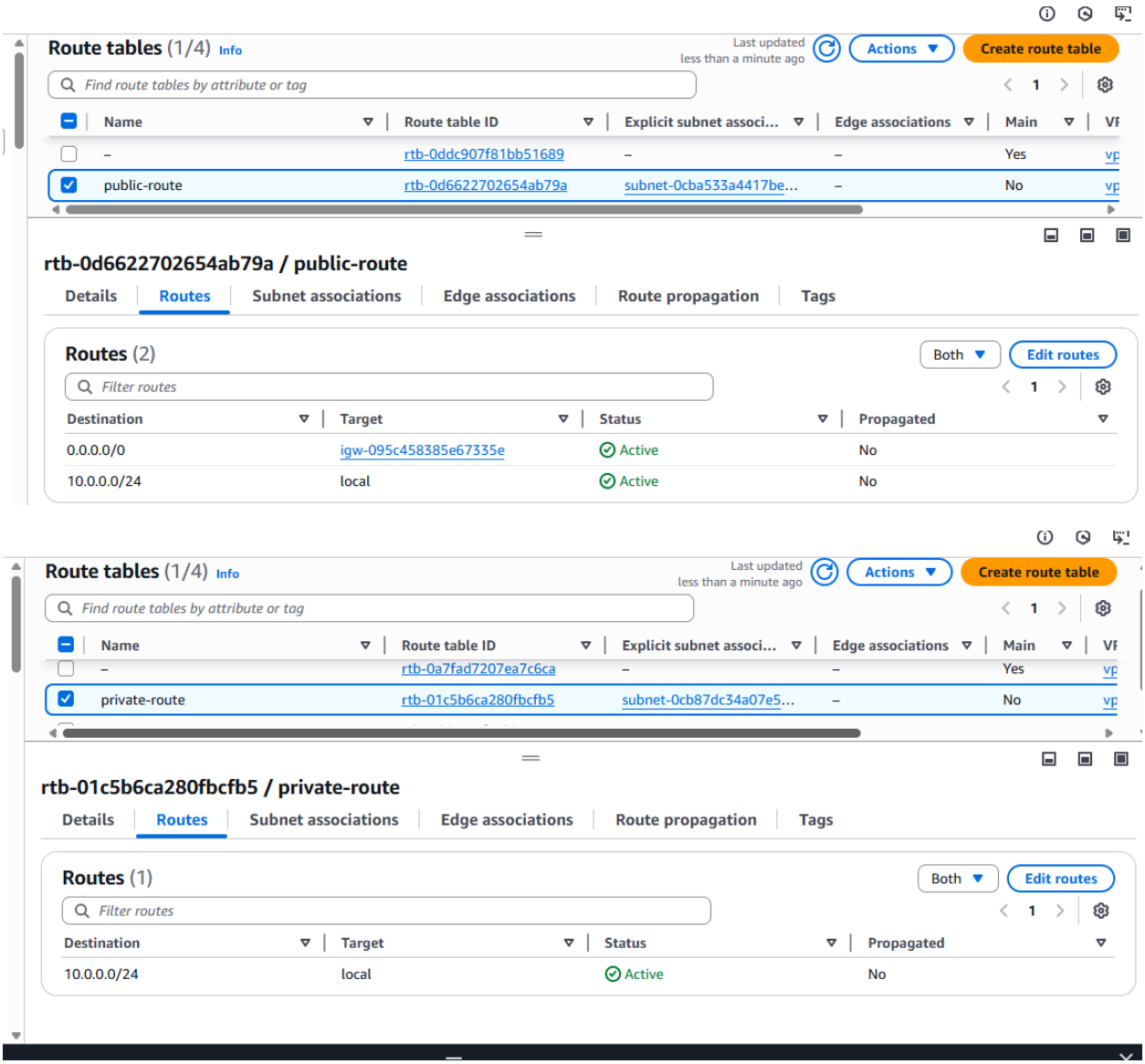




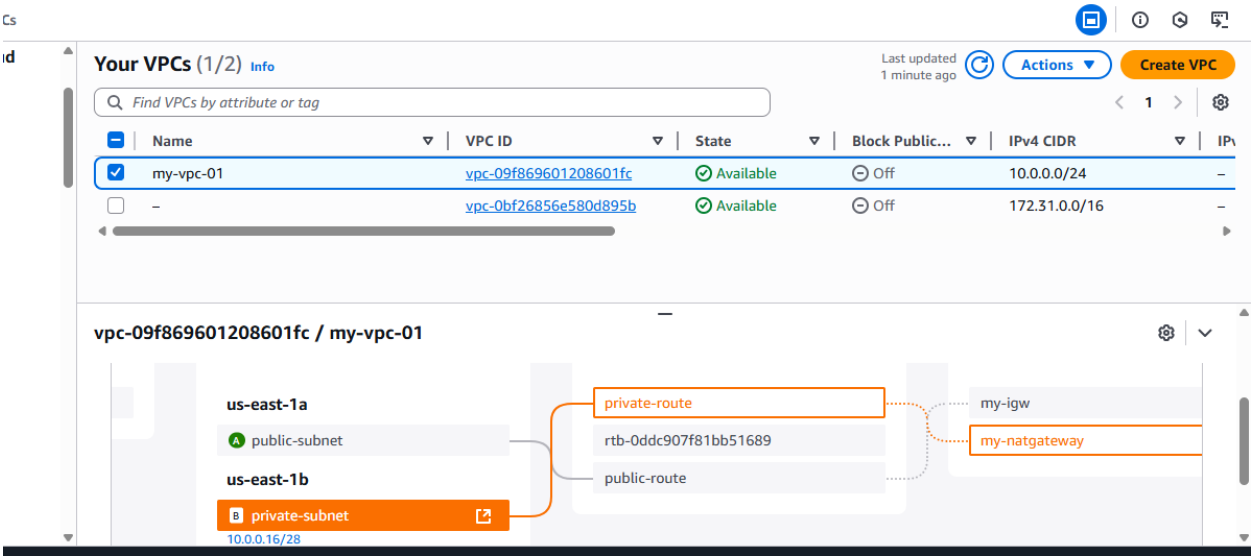
3) Provide the IGW to the Vpc.



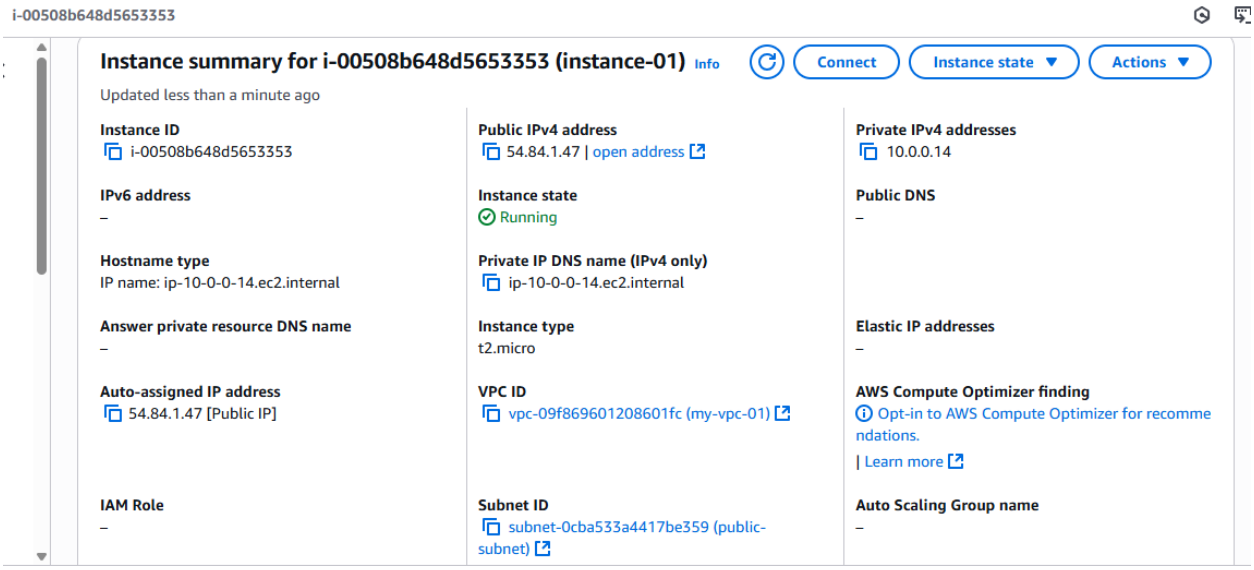
4) Create One public RT and one private RT.

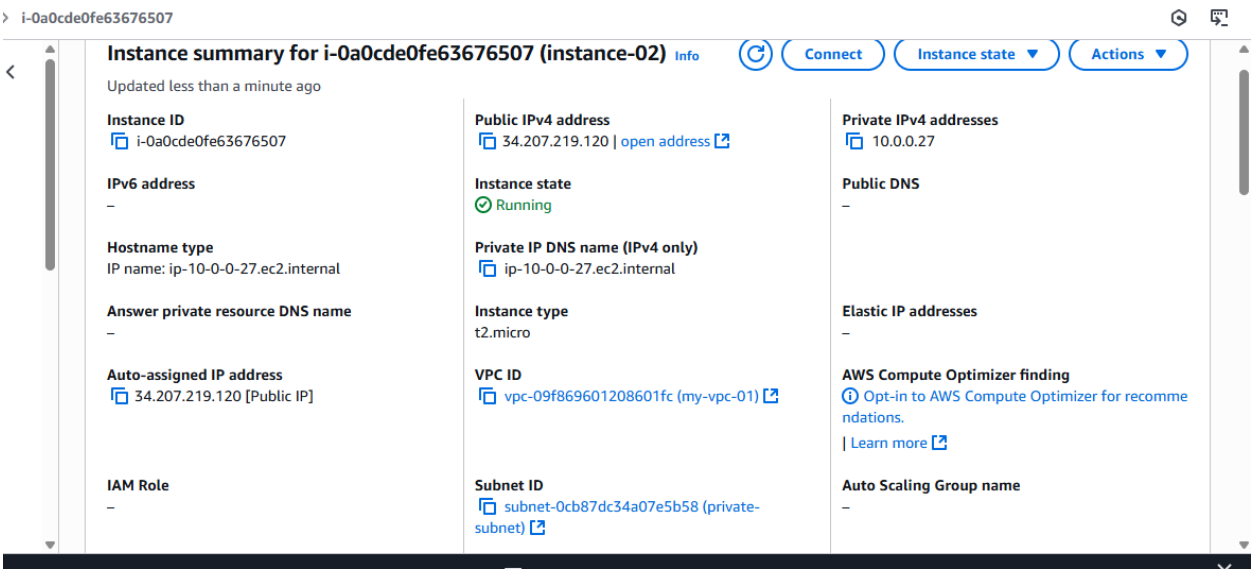


5) Deploy NAT gateway on public subnet and attach the NAT gateway to private subnet.



6) Create Two instances, one in public subnet and one in private subnet.



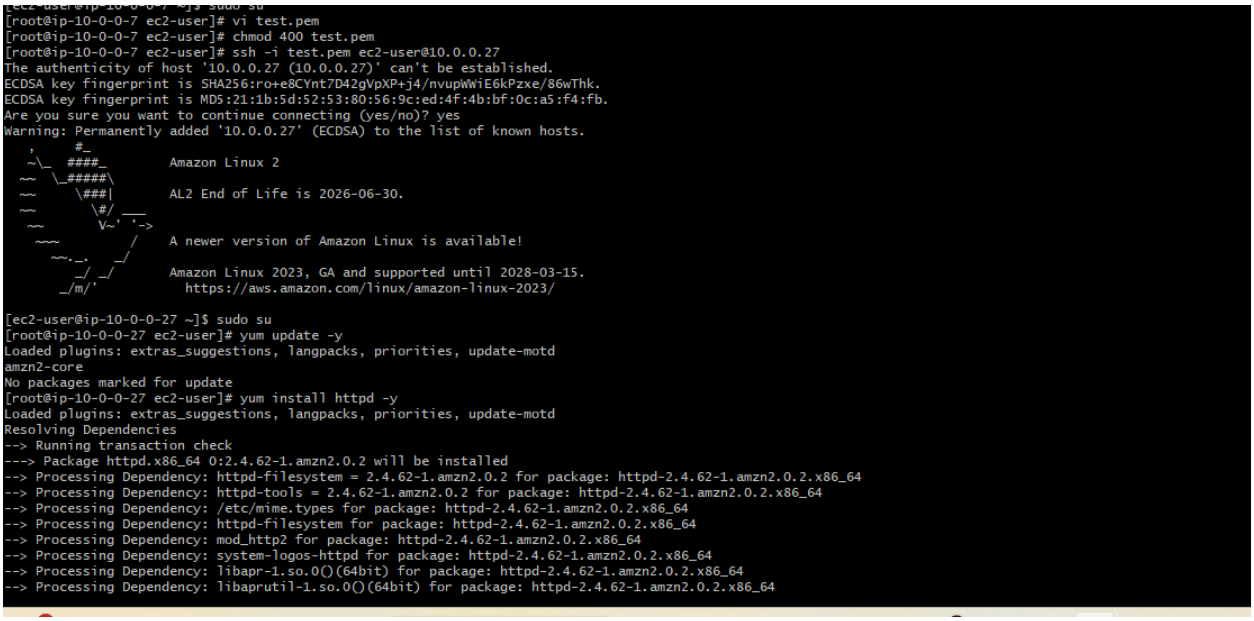


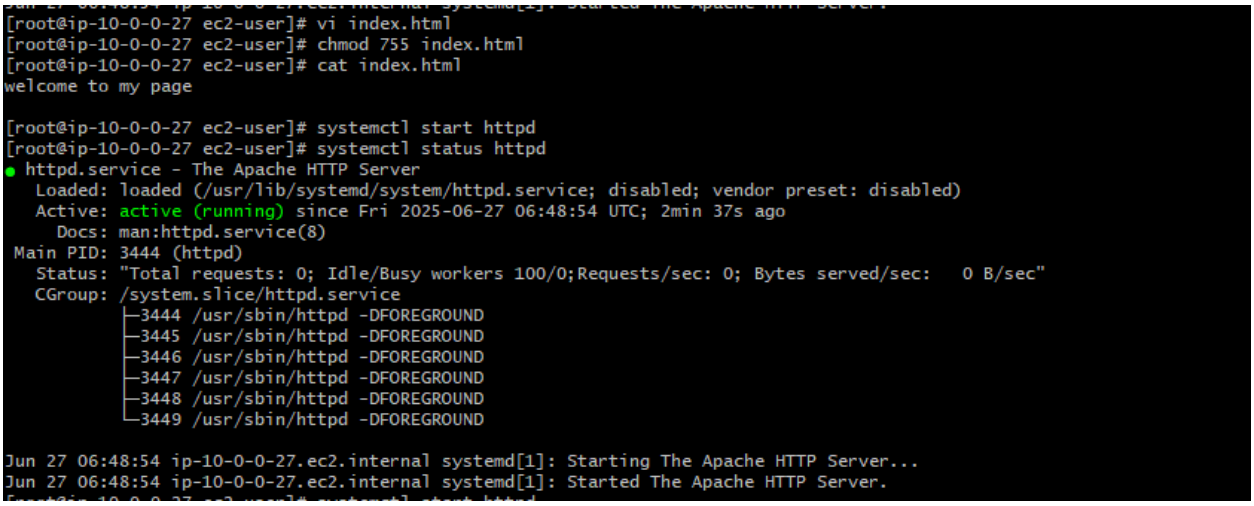
7) Deploy Apache server on both the ec2 instances with sample index.html file.

```
PC@DESKTOP-8IM0QCN MINGW64 ~/Downloads
$ ssh -i sushma.pem ec2-user@54.84.1.47
Last login: Thu Jun 26 13:39:50 2025 from 103.143.169.218
       #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-0-14 ~]$ sudo su
[root@ip-10-0-0-14 ec2-user]# vi index.html
[root@ip-10-0-0-14 ec2-user]# cd /var/www/html/
bash: cd: /var/www/html/: No such file or directory
[root@ip-10-0-0-14 ec2-user]# cd /var/www/html/
bash: cd: /var/www/html/: No such file or directory
[root@ip-10-0-0-14 ec2-user]# cd
[root@ip-10-0-0-14 ~]# sudo su
[root@ip-10-0-0-14 ~]#
[root@ip-10-0-0-14 ~]# sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
No packages marked for update
[root@ip-10-0-0-14 ~]# sudo yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.2-1.amzn2.0.1 will be installed
---> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.62-1.amzn2.0.2 will be installed
```

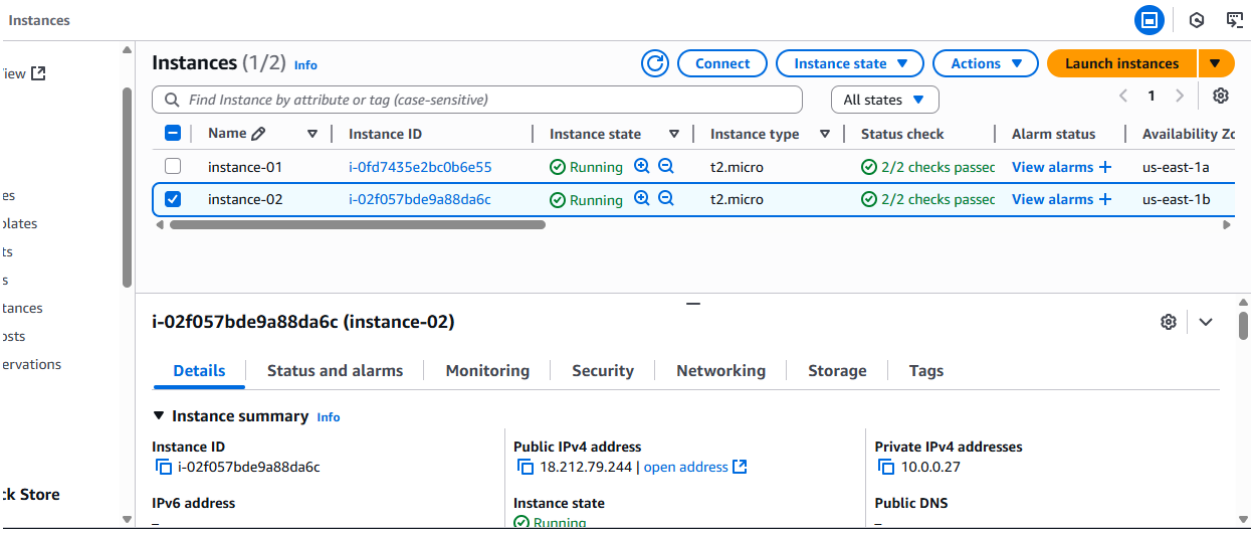
```
[root@ip-10-0-0-14 ~]# sudo systemctl start httpd
sudo systemctl enable httpd
[root@ip-10-0-0-14 ~]# sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[root@ip-10-0-0-14 ~]# echo "<h1>This is the Public EC2 Apache Page</h1>" | sudo tee /var/www/html/index.html
<h1>This is the Public EC2 Apache Page</h1>
[root@ip-10-0-0-14 ~]# vi index.html
[root@ip-10-0-0-14 ~]# chmod 755 index.html
[root@ip-10-0-0-14 ~]# cat index.html
welcome to my page

[root@ip-10-0-0-14 ~]# sudo systemctl start httpd
[root@ip-10-0-0-14 ~]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2025-06-26 13:57:45 UTC; 3min 25s ago
     Docs: man:httpd.service(8)
 Main PID: 1021 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─1021 /usr/sbin/httpd -DFOREGROUND
           ├─1025 /usr/sbin/httpd -DFOREGROUND
           ├─1026 /usr/sbin/httpd -DFOREGROUND
           ├─1027 /usr/sbin/httpd -DFOREGROUND
           ├─1028 /usr/sbin/httpd -DFOREGROUND
           └─1029 /usr/sbin/httpd -DFOREGROUND

Jun 26 13:57:45 ip-10-0-0-14.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Jun 26 13:57:45 ip-10-0-0-14.ec2.internal systemd[1]: Started The Apache HTTP Server.
[root@ip-10-0-0-14 ~]#
```

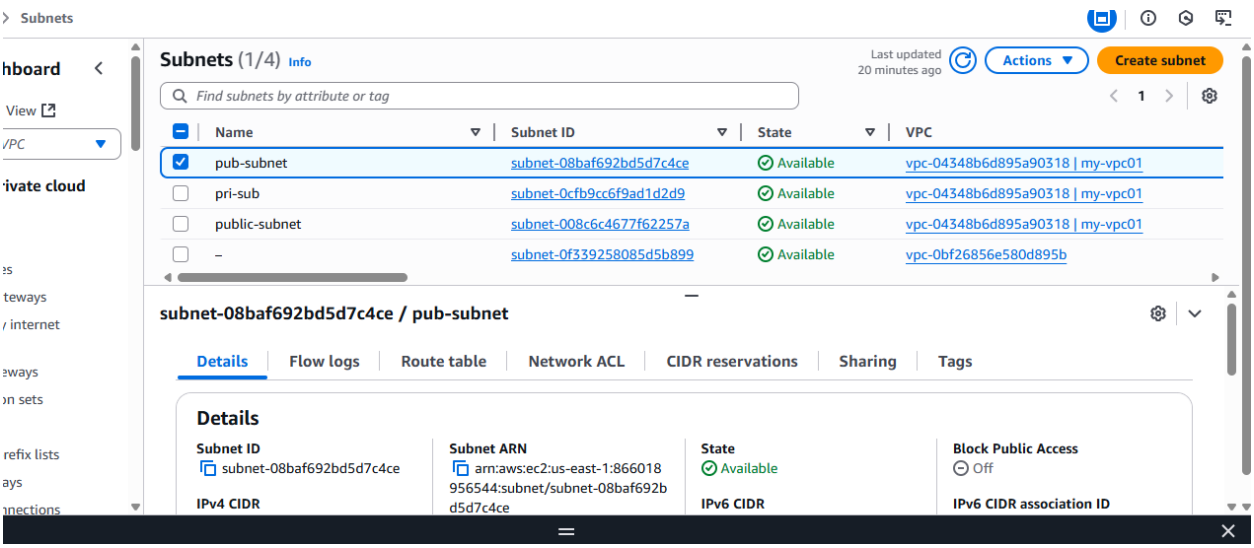
This is the Public EC2 Apache Page

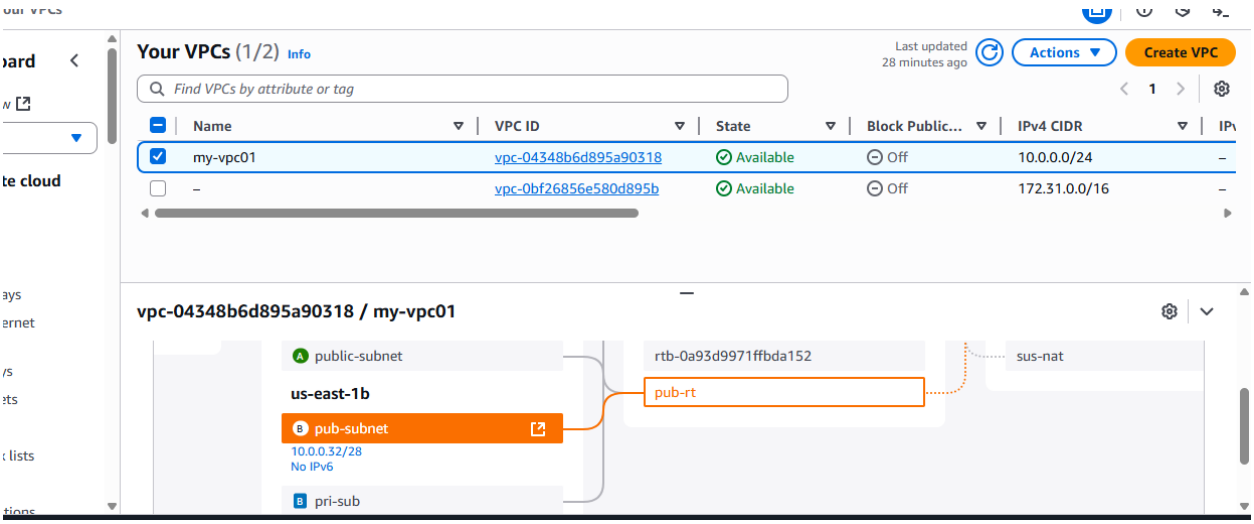


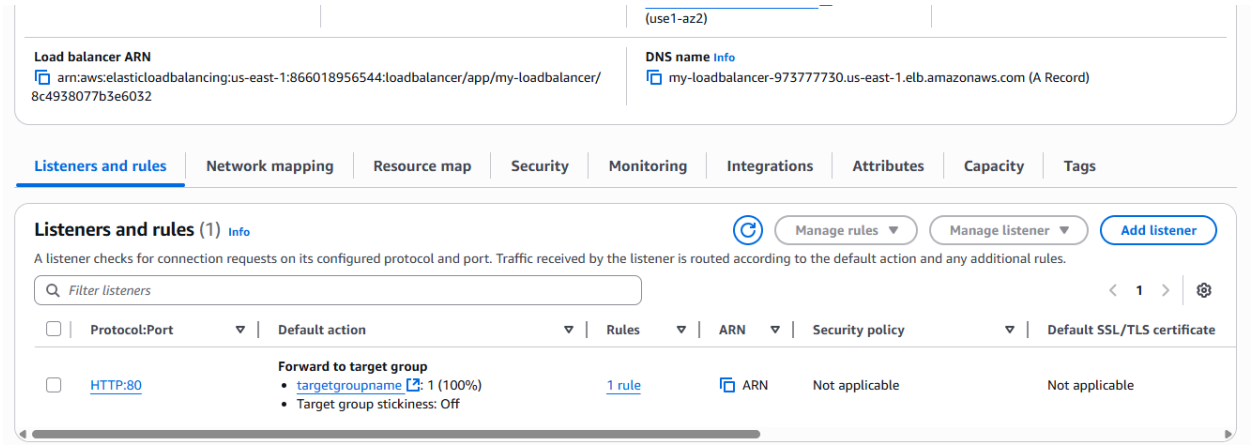


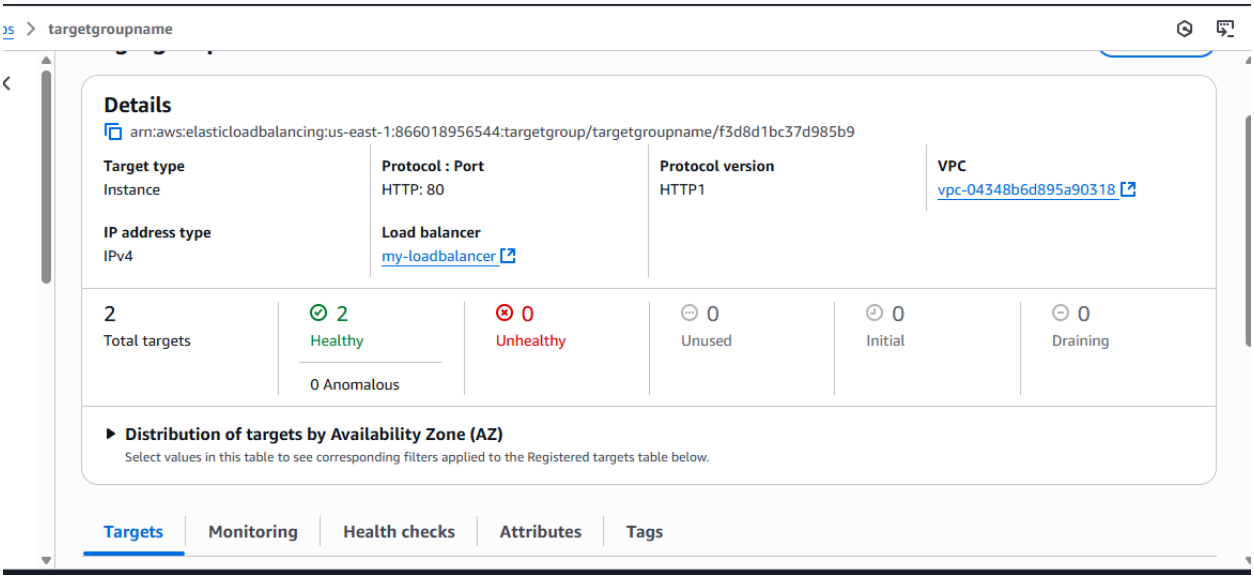
8) Create one application load balancer and attach the load balancer to both the ec2 instances.

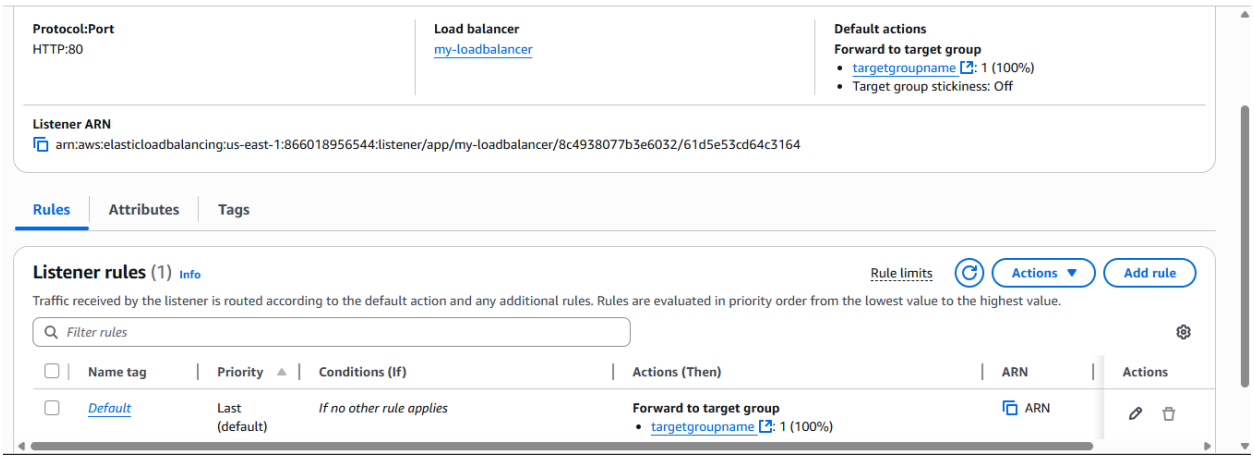

Instances
Instances (1/2) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zc
instance-01
i-0fd7435e2bc0b6e55
Running
t2.micro
2/2 checks passec
View alarms +
us-east-1a
instance-02
i-02f057bde9a88da6c
Running
t2.micro
2/2 checks passec
View alarms +
us-east-1b
i-02f057bde9a88da6c (instance-02)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-02f057bde9a88da6c
Public IPv4 address
18.212.79.244 | open address
Private IPv4 addresses
10.0.0.27
IPv6 address
Instance state
Running
Public DNS


Subnets
Subnets (1/4) Info
Last updated
20 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Name
Subnet ID
State
VPC
pub-subnet
subnet-08baf692bd5d7c4ce
Available
vpc-04348b6d895a90318 | my-vpc01
pri-sub
subnet-0cfb9cc6f9ad1d2d9
Available
vpc-04348b6d895a90318 | my-vpc01
public-subnet
subnet-008c6c4677f62257a
Available
vpc-04348b6d895a90318 | my-vpc01
–
subnet-0f339258085d5b899
Available
vpc-0bf26856e580d895b
subnet-08baf692bd5d7c4ce / pub-subnet
Details
Flow logs
Route table
Network ACL
CIDR reservations
Sharing
Tags
Details
Subnet ID
subnet-08baf692bd5d7c4ce
Subnet ARN
arn:aws:ec2:us-east-1:866018956544:subnet/subnet-08baf692bd5d7c4ce
State
Available
Block Public Access
Off
IPv4 CIDR
IPv6 CIDR
IPv6 CIDR association ID


Your VPCs (1/2) Info
Last updated
28 minutes ago
Actions
Create VPC
Find VPCs by attribute or tag
Name
VPC ID
State
Block Public...
IPv4 CIDR
my-vpc01
vpc-04348b6d895a90318
Available
Off
10.0.0.0/24
–
vpc-0bf26856e580d895b
Available
Off
172.31.0.0/16
vpc-04348b6d895a90318 / my-vpc01
public-subnet
us-east-1b
pub-subnet
10.0.0.32/28
No IPv6
pri-sub
rtb-0a93d9971ffbda152
pub-rt
sus-nat


(use1-az2)
Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:866018956544:loadbalancer/app/my-loadbalancer/8c4938077b3e6032
DNS name Info
my-loadbalancer-973777730.us-east-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
Integrations
Attributes
Capacity
Tags
Listeners and rules (1) Info
Manage rules
Manage listener
Add listener
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.
Filter listeners
Protocol:Port
Default action
Rules
ARN
Security policy
Default SSL/TLS certificate
HTTP:80
Forward to target group
targetgroupname: 1 (100%)
Target group stickiness: Off
1 rule
ARN
Not applicable
Not applicable

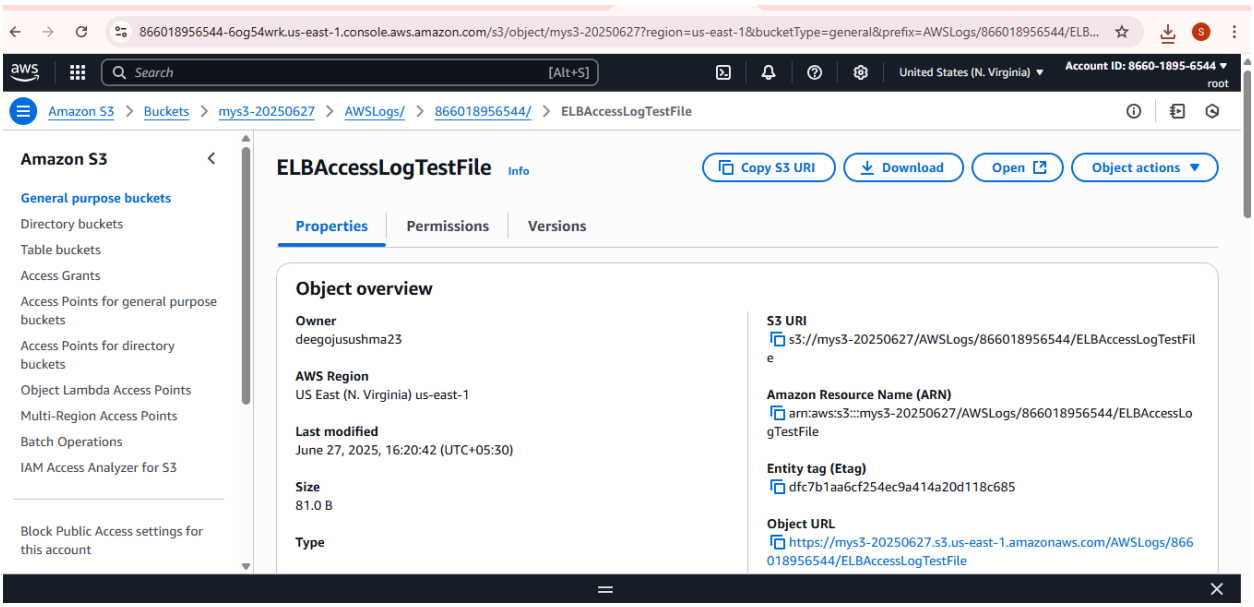

targetgroupname
Details
arn:aws:elasticloadbalancing:us-east-1:866018956544:targetgroup/targetgroupname/f3d8d1bc37d985b9
Target type
Instance
Protocol : Port
HTTP: 80
Protocol version
HTTP1
VPC
vpc-04348b6d895a90318
IP address type
IPv4
Load balancer
my-loadbalancer
2
Total targets
2
Healthy
0 Anomalous
0
Unhealthy
0
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags


Protocol:Port
HTTP:80
Load balancer
my-loadbalancer
Default actions
Forward to target group
targetgroupname: 1 (100%)
Target group stickiness: Off
Listener ARN
arn:aws:elasticloadbalancing:us-east-1:866018956544:listener/app/my-loadbalancer/8c4938077b3e6032/61d5e53cd64c3164
Rules
Attributes
Tags
Listener rules (1) Info
Rule limits
Actions
Add rule
Traffic received by the listener is routed according to the default action and any additional rules. Rules are evaluated in priority order from the lowest value to the highest value.
Filter rules
Name tag
Priority
Conditions (If)
Actions (Then)
ARN
Actions
Default
Last (default)
If no other rule applies
Forward to target group
targetgroupname: 1 (100%)
ARN

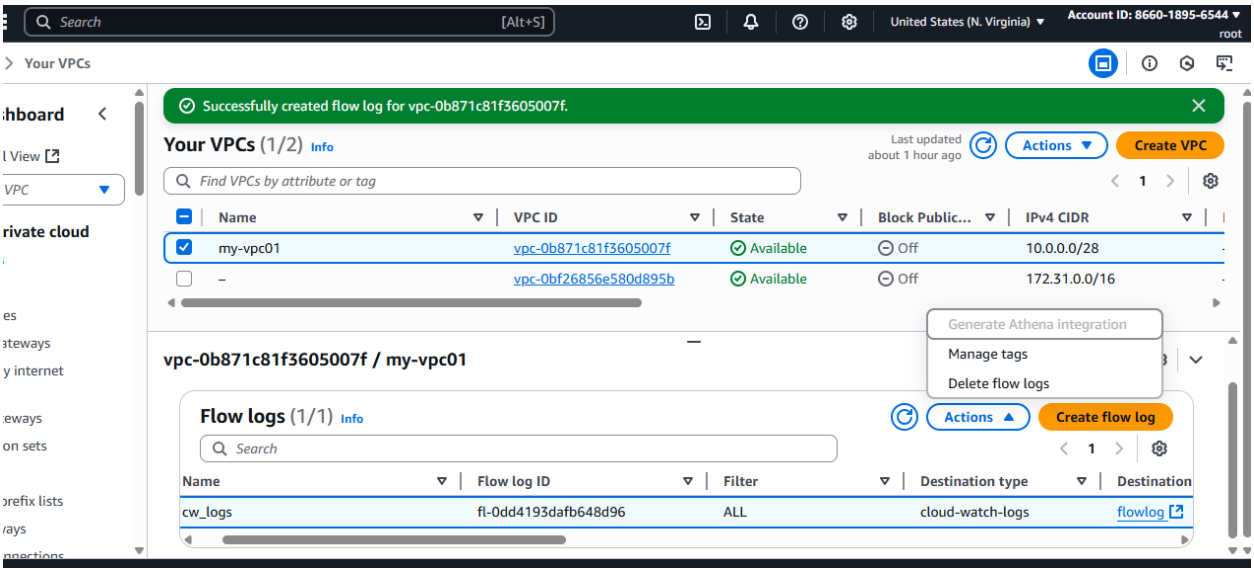
hello this is sushma website server one public server

hello private server sushma

9) Store Application load balancer logs to s3.

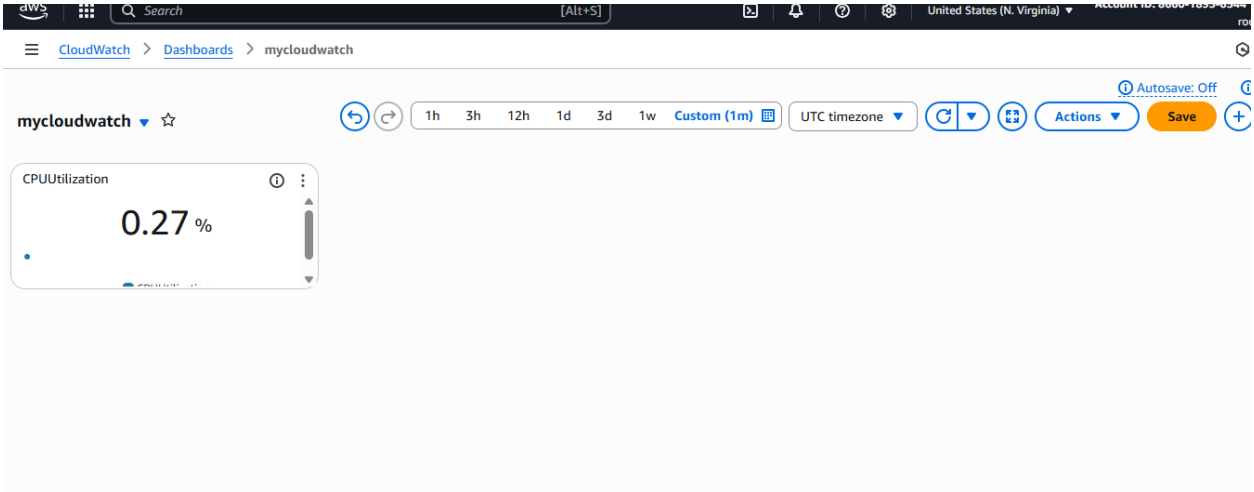


10) Store the Vpc flow logs to Cloudwatch group.

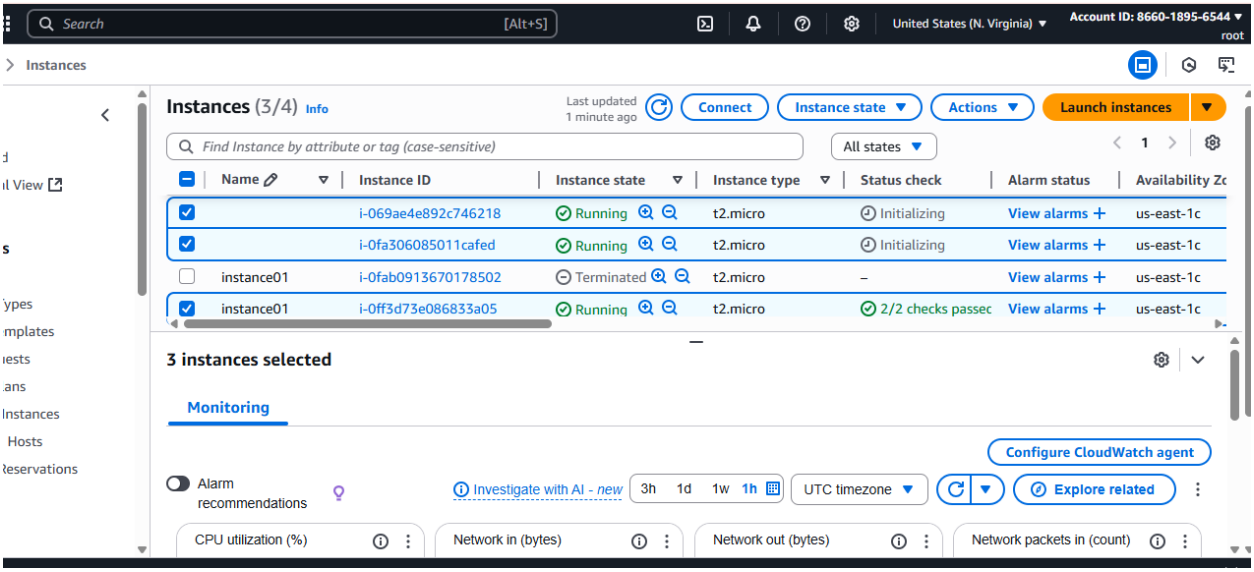


11) Create Monitoring Dashboards to monitor CPU utilization and to monitor Apache service.

```
Jun 27 12:54:23 ip-172-31-45-99.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Jun 27 12:54:23 ip-172-31-45-99.ec2.internal systemd[1]: Started The Apache HTTP Server.
[root@ip-172-31-45-99 ec2-user]# vi apache-monitor.sh
[root@ip-172-31-45-99 ec2-user]# chmod +x apache-monitor.sh
[root@ip-172-31-45-99 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2025-06-27 12:54:23 UTC; 38min ago
     Docs: man:httpd.service(8)
 Main PID: 3703 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─3703 /usr/sbin/httpd -DFOREGROUND
           ├─3704 /usr/sbin/httpd -DFOREGROUND
           ├─3705 /usr/sbin/httpd -DFOREGROUND
           ├─3706 /usr/sbin/httpd -DFOREGROUND
           ├─3707 /usr/sbin/httpd -DFOREGROUND
           └─3708 /usr/sbin/httpd -DFOREGROUND

Jun 27 12:54:23 ip-172-31-45-99.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Jun 27 12:54:23 ip-172-31-45-99.ec2.internal systemd[1]: Started The Apache HTTP Server.
[root@ip-172-31-45-99 ec2-user]# crontab -e
* * * * * /path/to/apache_monitor.sh
crontab: installing new crontab
[root@ip-172-31-45-99 ec2-user]# bash /path/to/apache_monitor.sh
bash: /path/to/apache_monitor.sh: No such file or directory
[root@ip-172-31-45-99 ec2-user]# vi apache-monitor.sh
[root@ip-172-31-45-99 ec2-user]# chmod +x apache-monitor.sh
[root@ip-172-31-45-99 ec2-user]# sudo bash /apache_monitor.sh
bash: /apache_monitor.sh: No such file or directory
[root@ip-172-31-45-99 ec2-user]# bash apache-monitor.sh
[root@ip-172-31-45-99 ec2-user]# crontab -e
crontab: installing new crontab
[root@ip-172-31-45-99 ec2-user]# /var/log/apache_monitor.log
bash: /var/log/apache_monitor.log: No such file or directory
[root@ip-172-31-45-99 ec2-user]# * * * * * /apache_monitor.sh >> /var/log/apache_monitor.log 2>&1
[root@ip-172-31-45-99 ec2-user]#
```



12) CPU utilizations more than 70% then it should trigger Autoscaling and launch new instance.


Search
[Alt+S]
United States (N. Virginia)
Account ID: 8660-1895-6544
root
Instances
Instances (3/4) Info
Last updated
1 minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zc
i-069ae4e892c746218 Running t2.micro Initializing View alarms us-east-1c
i-0fa306085011cafed Running t2.micro Initializing View alarms us-east-1c
instance01 i-0fab0913670178502 Terminated t2.micro – View alarms us-east-1c
instance01 i-0ff3d73e086833a05 Running t2.micro 2/2 checks passec View alarms us-east-1c
3 instances selected
Monitoring
Configure CloudWatch agent
Alarm recommendations
Investigate with AI - new
3h 1d 1w 1h
UTC timezone
Explore related
CPU utilization (%)
Network in (bytes)
Network out (bytes)
Network packets in (count)