

TASK ON AUTOSCALING GROUPS

1) Create one vpc in N.virginia region:

The screenshot shows the AWS VPC dashboard for the 'United States (N. Virginia)' region. The left sidebar contains navigation links for 'Virtual private cloud' and 'Security'. The main content area displays 'Your VPCs (1/1)' with a table listing the VPCs. The table has columns for Name, VPC ID, State, Block Public Access, IPv4 CIDR, and IPv6 CIDR. A single VPC named 'new vpc-vpc' with ID 'vpc-074ecc6bbf0a9e7be' is listed, with a state of 'Available' and 'Block Public Access' set to 'Off'. Below the table, the details for the selected VPC are shown, including its ID, state, block public access setting, and DNS hostnames.

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
new vpc-vpc	vpc-074ecc6bbf0a9e7be	Available	Off	10.0.0.0/16	-

vpc-074ecc6bbf0a9e7be / new vpc-vpc

Details

VPC ID	State	Block Public Access	DNS hostnames
vpc-074ecc6bbf0a9e7be	Available	Off	Disabled

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

Private Subnet:

The screenshot shows the AWS Subnets dashboard for the 'United States (N. Virginia)' region. The left sidebar contains navigation links for 'Virtual private cloud' and 'Security'. The main content area displays 'Subnets (1/2)' with a table listing the subnets. The table has columns for Name, Subnet ID, State, VPC, Block Public Access, and IPv4 CIDR. Two subnets are listed: 'private subnet' with ID 'subnet-03f22433b9525eb5d' and 'public subnet' with ID 'subnet-01b85ad6e57024108'. Both subnets are in the 'Available' state and have 'Block Public Access' set to 'Off'. Below the table, the details for the selected 'private subnet' are shown, including its ID, Subnet ARN, state, block public access setting, and IPv6 CIDR.

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
private subnet	subnet-03f22433b9525eb5d	Available	vpc-074ecc6bbf0a9e7be new ...	Off	10.0.128.0/20
public subnet	subnet-01b85ad6e57024108	Available	vpc-074ecc6bbf0a9e7be new ...	Off	10.0.0.0/20

subnet-03f22433b9525eb5d / private subnet

Details

Subnet ID	Subnet ARN	State	Block Public Access
subnet-03f22433b9525eb5d	arn:aws:ec2:us-east-1:971422718404:subnet/subnet-03f22433b9525eb5d	Available	Off
IPv4 CIDR		IPv6 CIDR	IPv6 CIDR association ID

Public Subnet

Subnets (1/2) Info

Find resources by attribute or tag

<input type="checkbox"/>	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
<input type="checkbox"/>	private subnet	subnet-03f22433b9525eb5d	Available	vpc-074ecc6bbf0a9e7be new ...	Off	10.0.128.0/20
<input checked="" type="checkbox"/>	public subnet	subnet-01b85ad6e57024108	Available	vpc-074ecc6bbf0a9e7be new ...	Off	10.0.0.0/20

subnet-01b85ad6e57024108 / public subnet

Details

Subnet ID subnet-01b85ad6e57024108	Subnet ARN arn:aws:ec2:us-east-1:971422718404:subnet/subnet-01b85ad6e57024108	State Available	Block Public Access Off
IPv4 CIDR 10.0.0.0/20		IPv6 CIDR -	IPv6 CIDR association ID -

3) Provide the IGW to the vpc:

Internet gateways (2) Info

Find resources by attribute or tag

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID	Owner
<input type="checkbox"/>	-	igw-09b1708710c095084	Detached	-	971422718404
<input checked="" type="checkbox"/>	new vpc-igw	igw-0bd775c11c1b27e10	Attached	vpc-074ecc6bbf0a9e7be new vpc-vpc	971422718404

new vpc-igw

Details

Internet gateway ID igw-0bd775c11c1b27e10	State Attached	VPC ID vpc-074ecc6bbf0a9e7be new vpc-vpc	Owner 971422718404
---	--------------------------	--	------------------------------

4) Create One public RT and one private RT:

Public RT:

Route tables (1/4) Info

Find resources by attribute or tag

<input type="checkbox"/>	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input type="checkbox"/>	-	rtb-0f0d716d90833545e	-	-	Yes	vpc-074ecc6bbf0a9e7be new ...
<input type="checkbox"/>	-	rtb-0c0d83107dcacd5e4	-	-	Yes	vpc-021be08e8e392964d
<input type="checkbox"/>	new vpc-rtb-private	rtb-049fe3bf18b4164f5	subnet-03f22433b9525e...	-	No	vpc-074ecc6bbf0a9e7be new ...
<input checked="" type="checkbox"/>	new vpc-rtb-public	rtb-00218224fa1856372	subnet-01b85ad6e57024...	-	No	vpc-074ecc6bbf0a9e7be new ...

rtb-00218224fa1856372 / new vpc-rtb-public

Details

Route table ID rtb-00218224fa1856372	Main No	Explicit subnet associations subnet-01b85ad6e57024108 / public subnet	Edge associations -
VPC vpc-074ecc6bbf0a9e7be new vpc-vpc	Owner ID 971422718404		

Private RT:

The screenshot shows the AWS Management Console interface for the 'Route tables' page. The left sidebar contains the 'VPC dashboard' and a list of resources under 'Virtual private cloud'. The main content area displays a table of route tables. The 'new vpc-rtb-private' route table is selected, and its details are shown below the table. The details include the route table ID, main status, explicit subnet associations, and edge associations.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
-	rtb-0f0d716d90833545e	-	-	Yes	vpc-074ecc6bbf0a9e7be new
-	rtb-0c0d83107dcacd6e4	-	-	Yes	vpc-021be08e8e392964d
new vpc-rtb-private	rtb-049fe3bf18b4164f5	subnet-03f22433b9525e...	-	No	vpc-074ecc6bbf0a9e7be new
new vpc-rtb-public	rtb-00218224fa1856372	subnet-01b85ad6e57024...	-	No	vpc-074ecc6bbf0a9e7be new

Details

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-049fe3bf18b4164f5	No	subnet-03f22433b9525e5d / private subnet	-

VPC

VPC	Owner ID
vpc-074ecc6bbf0a9e7be new vpc-vpc	971422718404

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

→ Created NAT gateway on public subnet:

The screenshot shows the AWS Management Console interface for the 'NAT gateways' page. The left sidebar contains the 'VPC dashboard' and a list of resources under 'Virtual private cloud'. The main content area displays a table of NAT gateways. The 'nat-gateway-1' NAT gateway is selected, and its details are shown below the table. The details include the NAT gateway ID, connectivity type, state, and state message.

Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primary private I..
nat-gateway-1	nat-04cbb6ff22fba337d	Public	Available	-	54.163.89.245	10.0.6.37

Details

NAT gateway ID	Connectivity type	State	State message
nat-04cbb6ff22fba337d	Public	Available	-

→ Attached NAT gateway to Private subnet:

The screenshot shows the AWS VPC console with a green notification bar at the top stating: "Updated routes for rtb-049fe3bf18b4164f5 / new vpc-rtb-private successfully". Below this, the page title is "rtb-049fe3bf18b4164f5 / new vpc-rtb-private".

Details

Route table ID rtb-049fe3bf18b4164f5	Main No	Explicit subnet associations subnet-03f22433b9525eb5d / private subnet	Edge associations -
VPC vpc-074ecc6bbf0a9e7be new vpc-vpc	Owner ID 971422718404		

Routes (3)

Destination	Target	Status	Propagated
pl-63a5400a	vpc-07e2b5d54d8807a1a	Active	No
0.0.0.0/0	igw-0bdf75c11c1b27e10	Active	No
10.0.0.0/16	local	Active	No

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

The screenshot shows the AWS EC2 console with a list of instances. Two instances are shown:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
new vpc-ec2-2	i-08d8e9ac5d6e737b7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
new vpc-ec2-1	i-096c1e6163aa1325e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-

i-08d8e9ac5d6e737b7 (new vpc-ec2-2)

Instance summary

Instance ID i-08d8e9ac5d6e737b7	Public IPv4 address 34.205.48.252 open address	Private IPv4 addresses 10.0.3.224
IPv6 address -	Instance state Running	Public IPv4 DNS -

The screenshot shows the AWS EC2 console with a list of instances. Two instances are shown:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
new vpc-ec2-2	i-08d8e9ac5d6e737b7	Running	t2.micro	Initializing	View alarms +	us-east-1a	-
new vpc-ec2-1	i-096c1e6163aa1325e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-

i-096c1e6163aa1325e (new vpc-ec2-1)

Instance summary

Instance ID i-096c1e6163aa1325e	Public IPv4 address 34.237.1.75 open address	Private IPv4 addresses 10.0.3.224
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-3-224.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-3-224.ec2.internal	

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

EC1

Allow tags in metadata | Info
Select

User data - optional | Info
Upload a file with your user data or enter it in the field.
[Choose file](#)

```
#!/bin/bash
sudo yum -y install httpd
echo "welcome to lb-server-1" >> /var/www/html/index.html
sudo systemctl start httpd
```

☐ User data has already been base64 encoded

Summary
Number of instances | Info
1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-0f2ce9ce760bd7133

Virtual server type (instance type)
t2.micro

Firewall (security group)
default

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

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

← → ↻ ⚠ Not secure 13.233.102.122 ☆ 📄 🧑🏿 👤 ⋮

welcome to lb-server-1

EC2

Allow tags in metadata | Info
Select

User data - optional | Info
Upload a file with your user data or enter it in the field.
[Choose file](#)

```
#!/bin/bash
sudo yum -y install httpd
echo "welcome to lb-server-1" >> /var/www/html/index.html
sudo systemctl start httpd
```

☐ User data has already been base64 encoded

Summary
Number of instances | Info
1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-0f2ce9ce760bd7133

Virtual server type (instance type)
t2.micro

Firewall (security group)
default

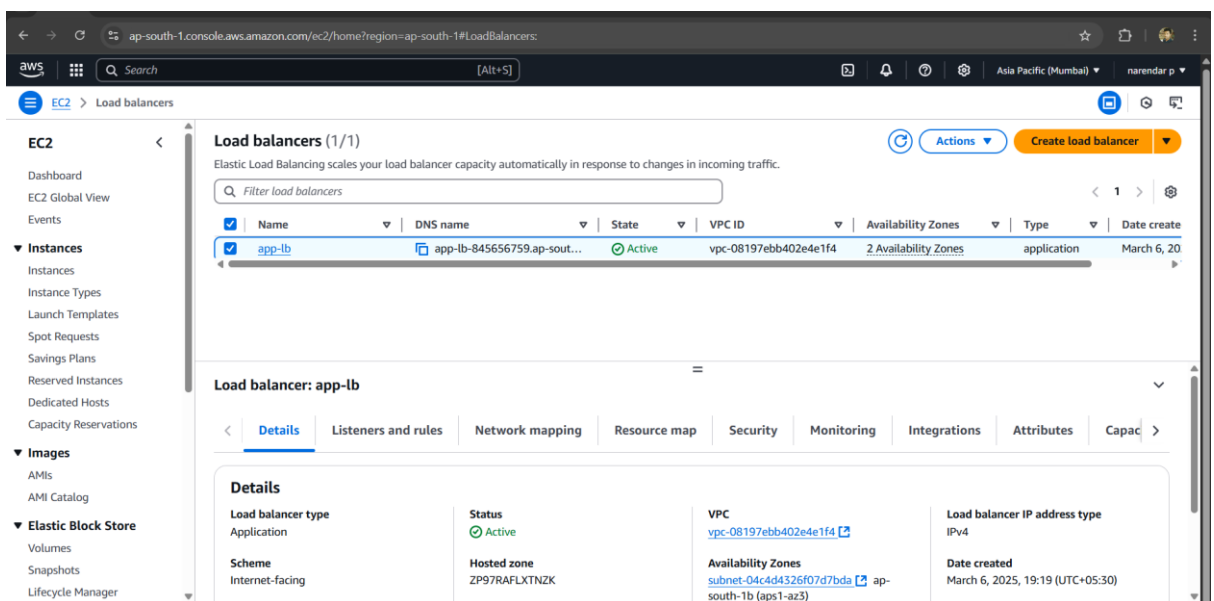
Storage (volumes)
1 volume(s) - 8 GiB

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

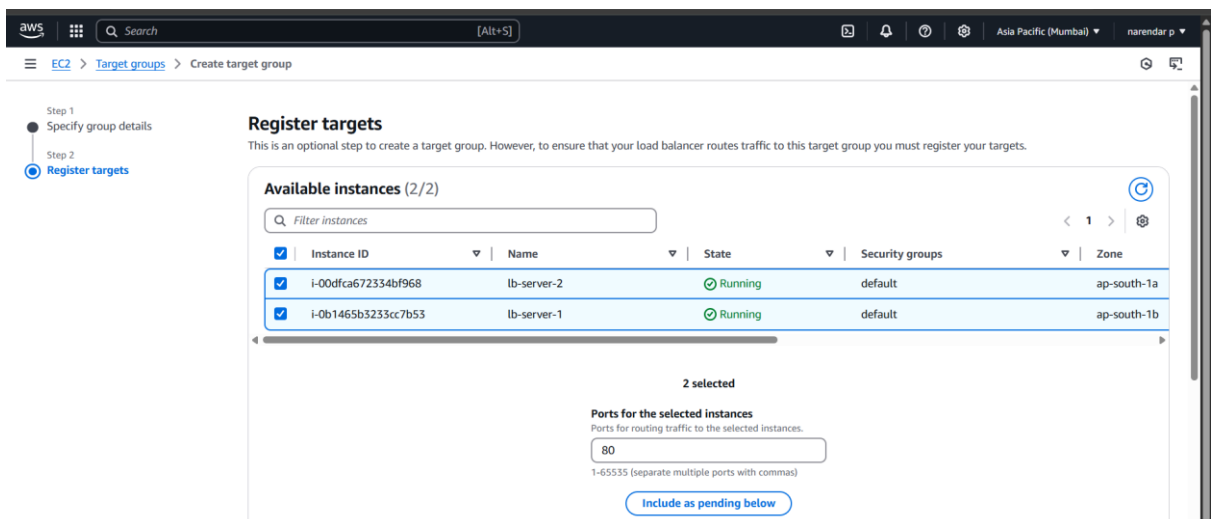


8) Create one application load balancer and attach the load balancer to both the ec2 instances:

→ Created Application Load balancer:

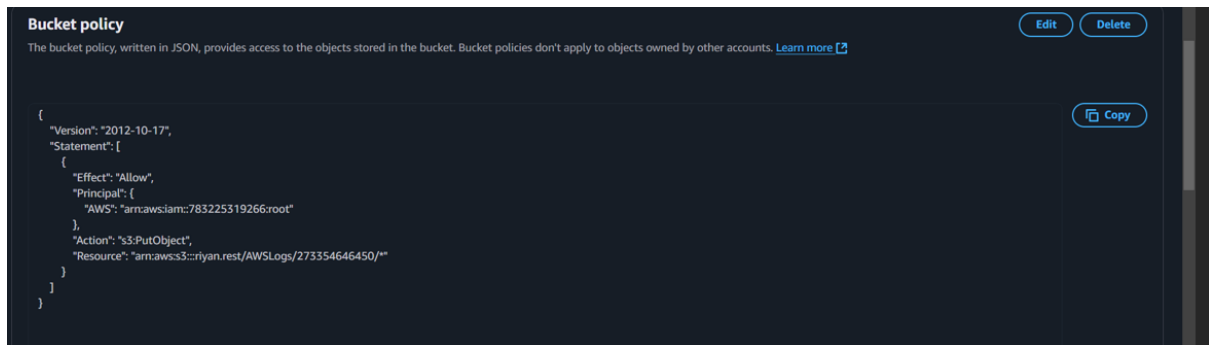


→ Attached The load balancer to both the Ec2 instances which are created above task:

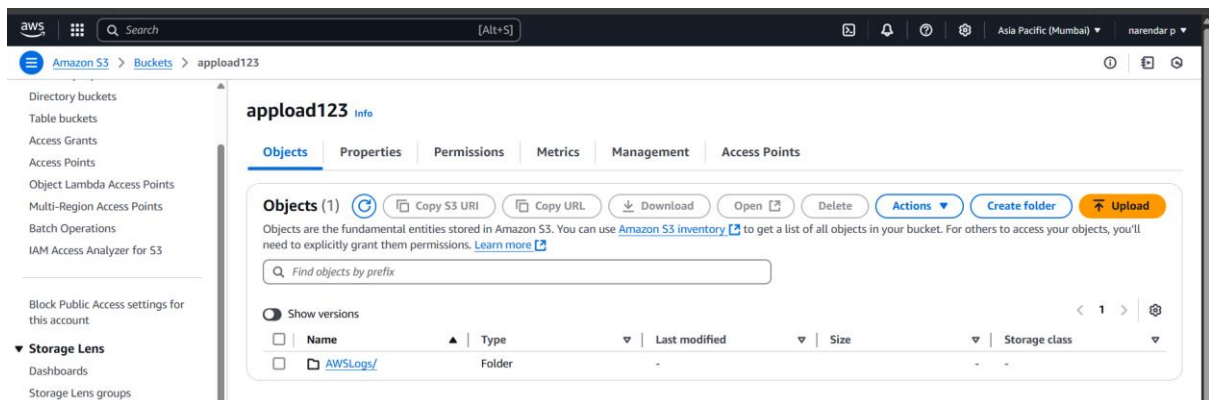


9) Store Application load balancer logs to s3:

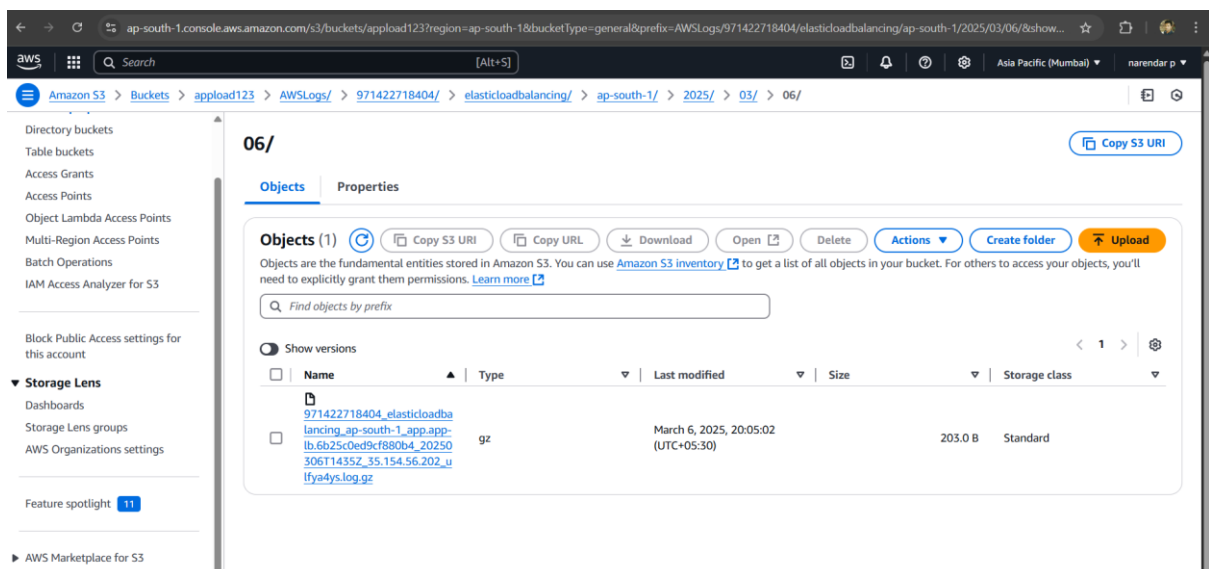
→ Created bucket policy:



→ Added application load balancer logs to s3:

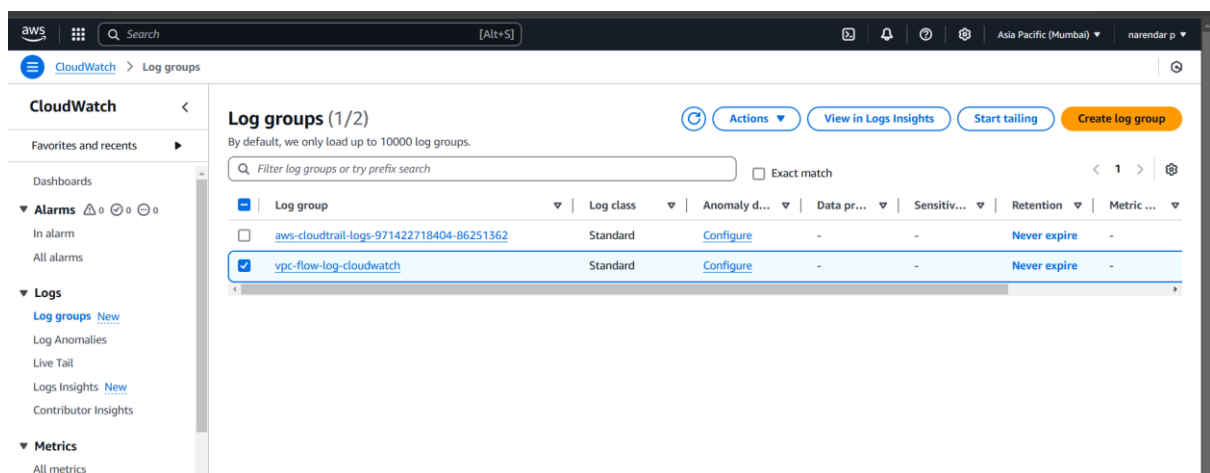


→ logs generated in s3:

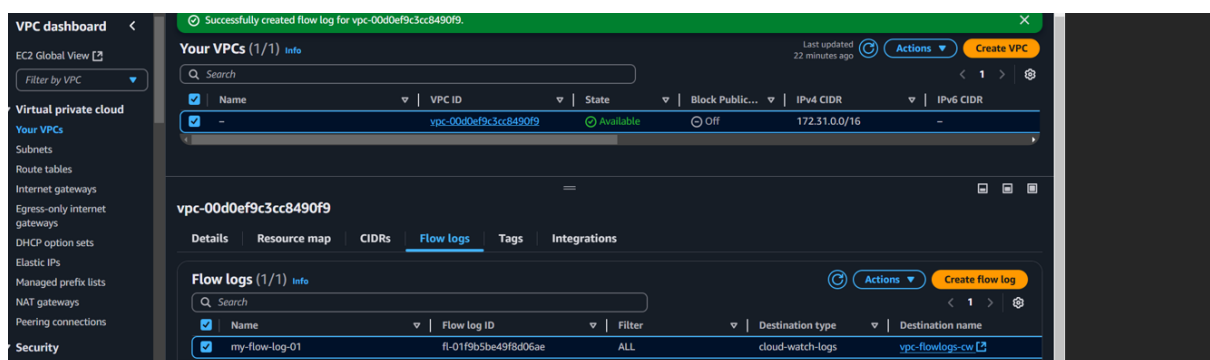


10) Store the vpc flow logs to cloud watch group:

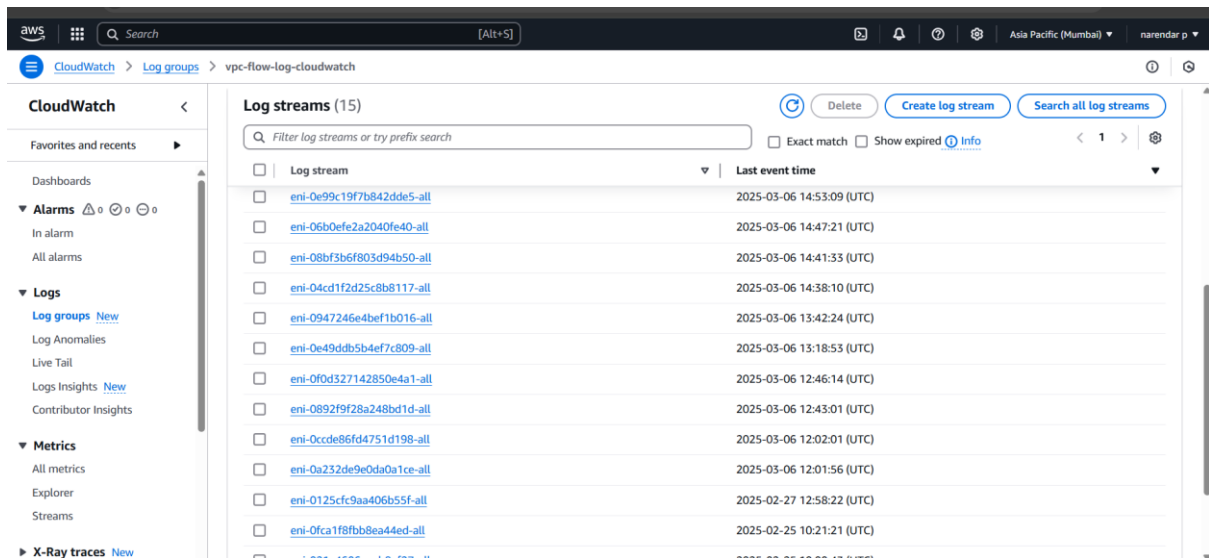
→Created log group in cloud watch:



→Created flow logs from VPC by giving the log group which we created



→The vpc logs generated in cloudwatch under log groups section:



11) Create Monitoring Dashboards to monitor cpu utilization and to monitor apache service:

→ Installed apache httpd in an instance:

```

Unit apache.service could not be found.
ec2-user@ip-10-0-0-200 ~]$ systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Wed 2025-03-05 10:14:26 UTC; 3h 30min ago
     Docs: man:httpd.service(8)
  Main PID: 3446 (httpd)
    Status: "Total requests: 17; Idle/Busy workers 100/0; Requests/sec: 0.00134; Bytes served/sec: 1 B/sec"
    CGroup: /system.slice/httpd.service
            └─3446 /usr/sbin/httpd -DFOREGROUND
              └─3447 /usr/sbin/httpd -DFOREGROUND
                └─3448 /usr/sbin/httpd -DFOREGROUND
                  └─3449 /usr/sbin/httpd -DFOREGROUND
                    └─3450 /usr/sbin/httpd -DFOREGROUND
                      └─3451 /usr/sbin/httpd -DFOREGROUND
                        └─3506 /usr/sbin/httpd -DFOREGROUND
                          └─3572 /usr/sbin/httpd -DFOREGROUND

Mar 05 10:14:26 ip-10-0-0-200.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Mar 05 10:14:26 ip-10-0-0-200.ec2.internal systemd[1]: Started The Apache HTTP Server.

```

→script for apache monitoring:

```

#!/bin/bash
#set -x

# Get instance ID
INSTANCE_ID=$(/opt/aws/bin/ec2-metadata -i | awk '{print $2}')

# Function to check httpd process status
checkHttpdStatus() {
    ps x | grep 'httpd' | grep -v grep | wc -l
}

# Get httpd process count
i=$(checkHttpdStatus)

# Send metric to CloudWatch
if [ "$i" -eq 0 ]; then
    aws --region us-east-1 cloudwatch put-metric-data --metric-name httpd --value 0 --namespace h
    ttpd --dimensions InstanceId="$INSTANCE_ID"
else
    aws --region us-east-1 cloudwatch put-metric-data --metric-name httpd --value 1 --namespace h
    ttpd --dimensions InstanceId="$INSTANCE_ID"
fi

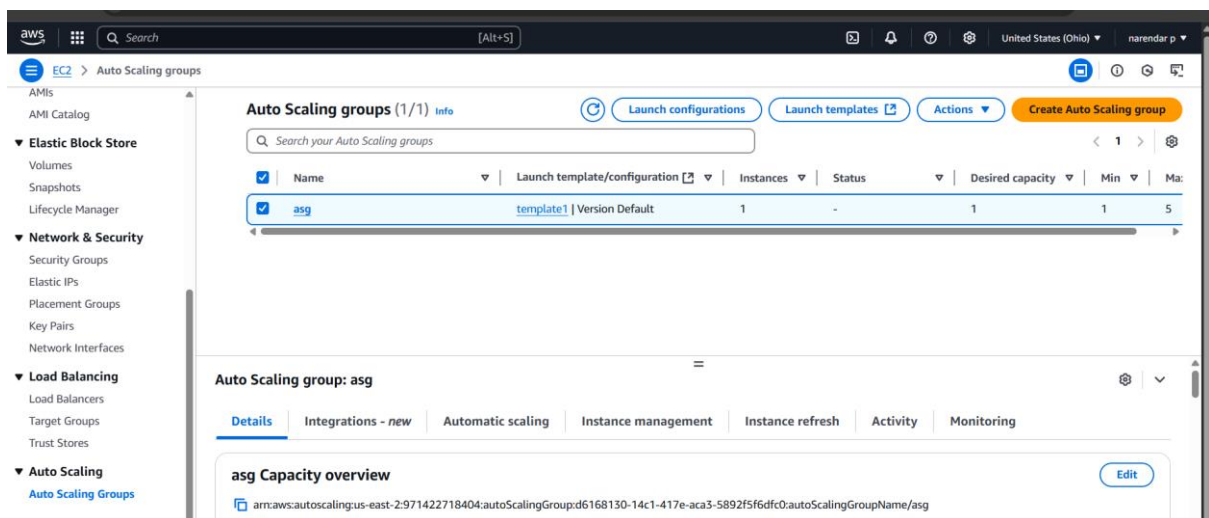
```

→created dashboard for cpu utilization:



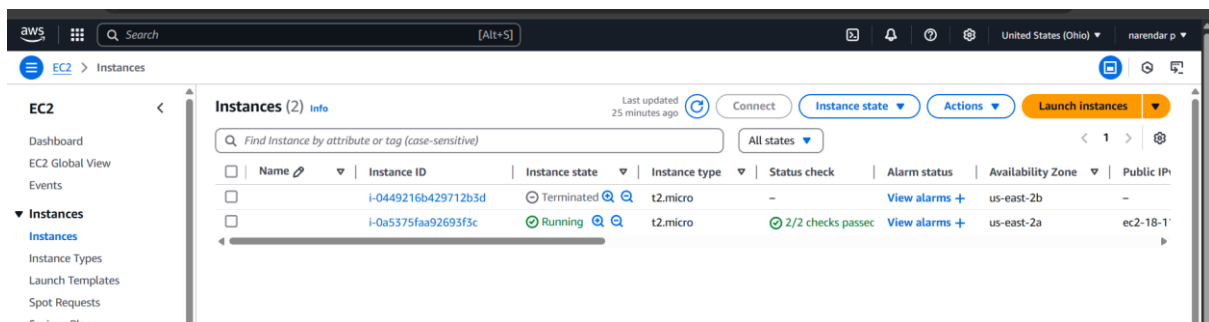
12) CPU utilizations more than 70% then it should trigger Autoscaling and launch new instance:

→ Created an AutoScaling group and attached to a template:



→ Created an Auto Scaling Group and defined scaling policies:

→ Instance terminated after stress applied:



→ Capacity added on unhealthy condition (more than 70% cpu utilization)

Activity history (3)

Status	Description	Cause	Start time
Successful	Terminating EC2 instance: i-0449216b429712b3d	At 2025-03-07T08:16:00Z a monitor alarm TargetTracking-asg-AlarmLow-7c4ff6cb-4710-48b3-9c7b-f8cbd670d52e in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 2 to 1. At 2025-03-07T08:16:03Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 2 to 1. At 2025-03-07T08:16:04Z instance i-0449216b429712b3d was selected for termination.	2025 March 07, 01:46:04 PM +05:30
Successful	Launching a new EC2 instance: i-0a5375faa92693f3c	At 2025-03-07T06:37:32Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2025-03-07T06:37:35Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2025 March 07, 12:07:36 PM +05:30
Successful	Launching a new EC2 instance: i-0449216b429712b3d	At 2025-03-07T06:37:32Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2025-03-07T06:37:35Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2025 March 07, 12:07:36 PM +05:30

→cpu utilization:

CPU utilization (%)

99%

Graphed metrics (1)

Label	Details	Statistic	Period	Y axis	Actions
i-0a5375faa92693f3c	Region: us-east-2 • EC2 • CPUUtilization	Average	1 minute		