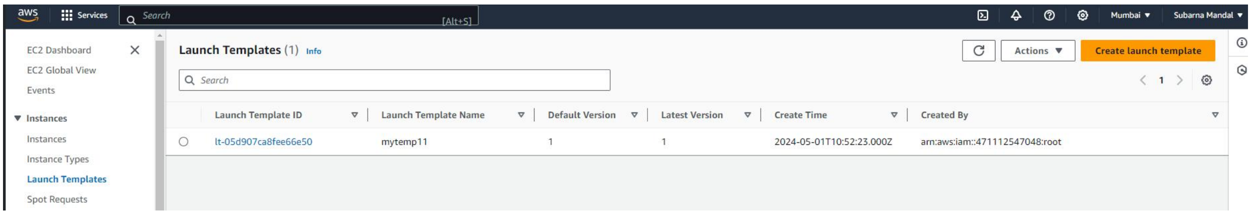
Assignment 11

Problem Statement: Build scaling plans in AWS that balance load on different EC2 instance

Steps:

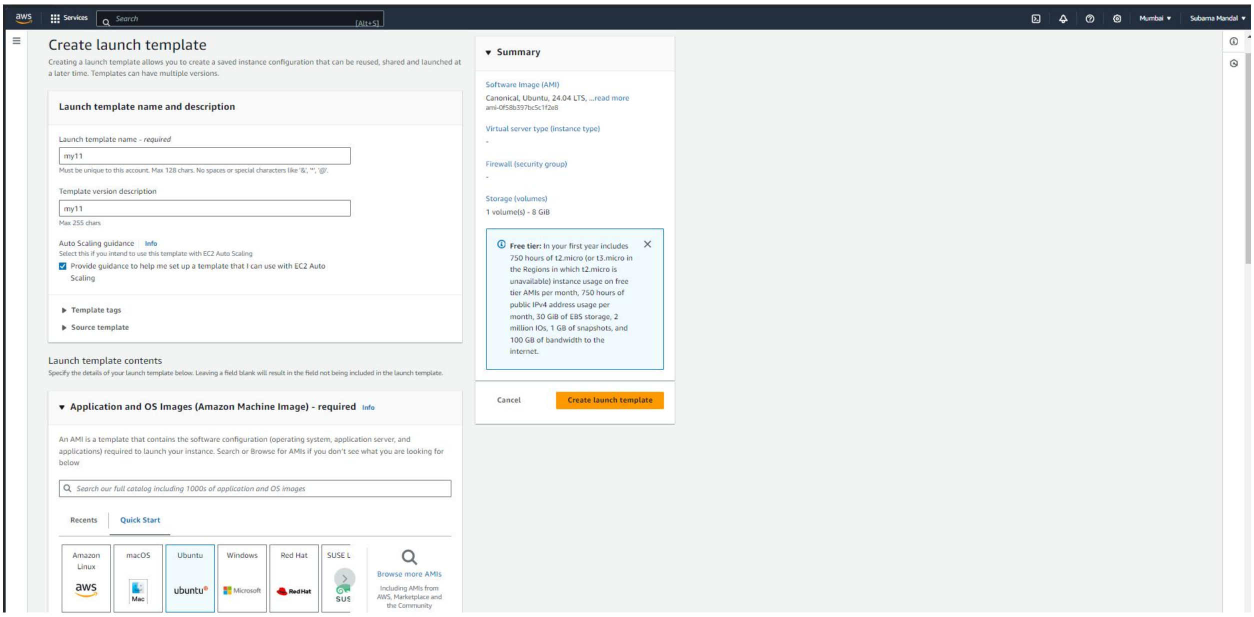
1. Open AWS and click EC2, next, select "Launch Template" from the menu on the left side. Now click on the "Create Launch Template" option.



Launch Templates (1) Info

	Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By
○	lt-05d907ca8fee66e50	mytemp11	1	1	2024-05-01T10:52:23.000Z	arn:aws:iam::471112547048:root

2. Enter a template name, such as "my11" and check the box for autoscaling option Navigate to "Quick Start" and choose "Ubuntu" from the list of available AMIs.



Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

my11

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

my11

Max 255 chars

Auto Scaling guidance Info

Select this if you intend to use this template with EC2 Auto Scaling

☑ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

▸ Template tags

▸ Source template

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

▾ Application and OS Images (Amazon Machine Image) - required Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

▾ Summary

Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-0f58b397bc5c1f2e8

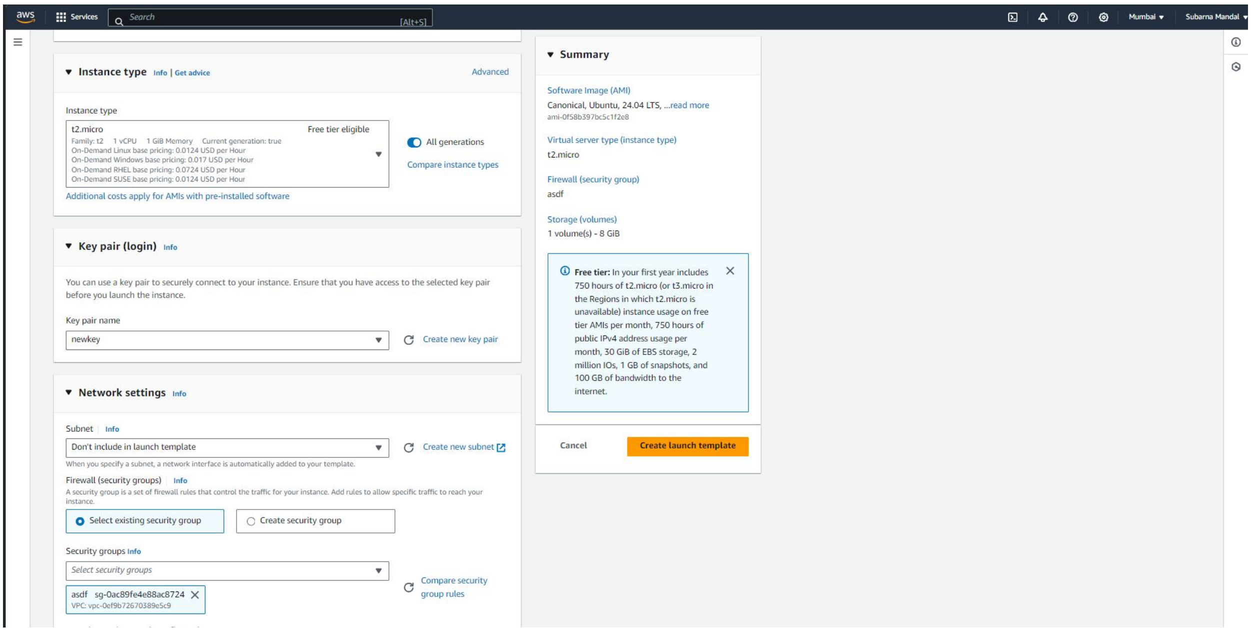
Virtual server type (instance type)
-

Firewall (security group)
-

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

3. Next, select the instance type - either t2.micro or t3.micro, both of which are free tier eligible. Then choose either an existing key pair or create a new one if it doesn't exist. Select an existing security group.



▾ Instance type Info | Get advice

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0724 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations

Additional costs apply for AMIs with pre-installed software

▾ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name

newkey

▾ Network settings Info

Subnet Info

Don't include in launch template

When you specify a subnet, a network interface is automatically added to your template.

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.

● Select existing security group ○ Create security group

Security groups Info

Select security groups

asdf sg-0ac89fe4e88ac8724
VPC: vpc-0ef9b72670389e5c9

▾ Summary

Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-0f58b397bc5c1f2e8

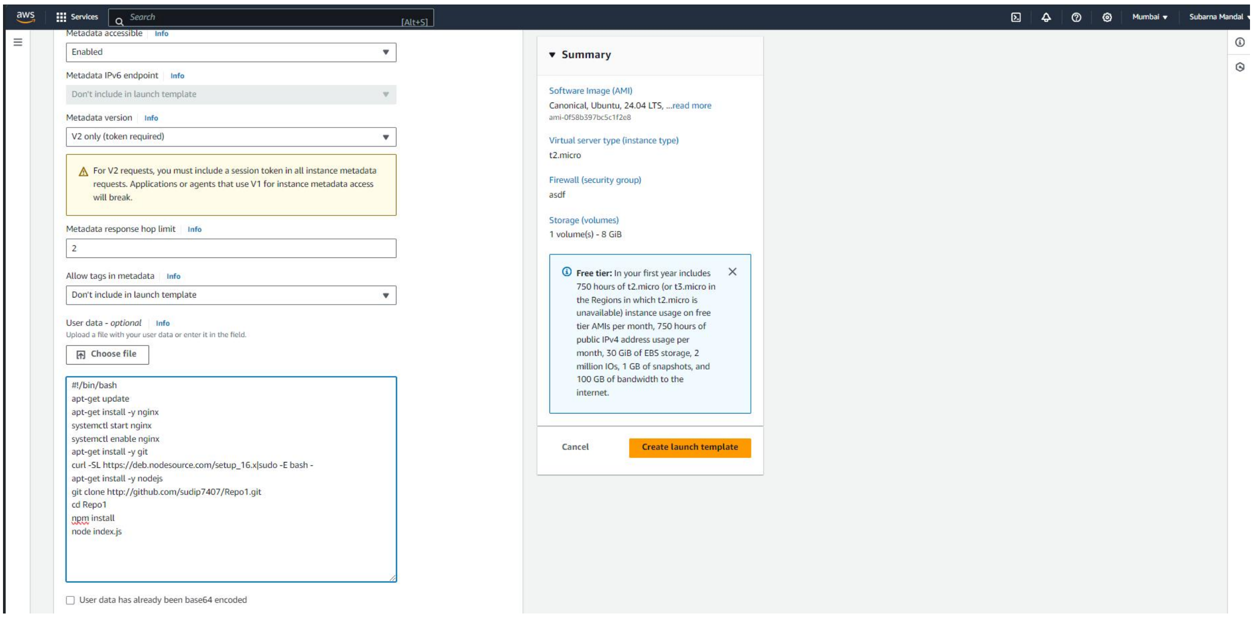
Virtual server type (instance type)
t2.micro

Firewall (security group)
asdf

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

4. Expand the "Advanced details" section, navigate to "User data", and input the provided code. Then proceed to click on “Create launch template”



Metadata accessible Info

Enabled

Metadata IPv6 endpoint Info

Don't include in launch template

Metadata version Info

V2 only (token required)

For V2 requests, you must include a session token in all instance metadata requests. Applications or agents that use V1 for instance metadata access will break.

Metadata response hop limit Info

2

Allow tags in metadata Info

Don't include in launch template

User data - *optional* Info

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
apt-get update
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
apt-get install -y git
curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash -
apt-get install -y nodejs
git clone http://github.com/sudip7407/Repo1.git
cd Repo1
npm install
node index.js
```

□ User data has already been base64 encoded

▼ Summary

Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-0f58b397bc5c1f2e8

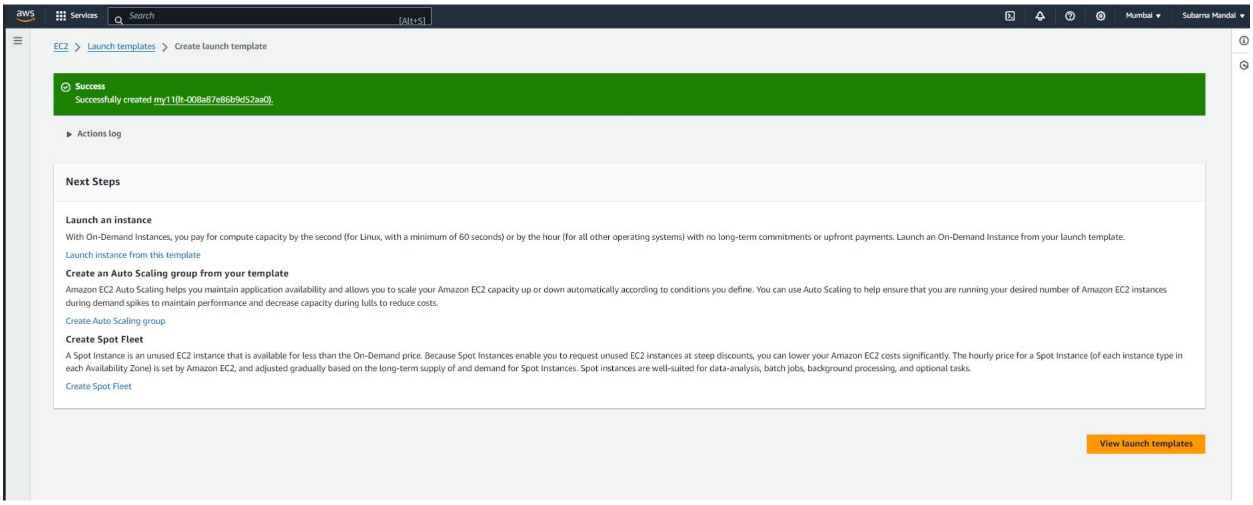
Virtual server type (instance type)
t2.micro

Firewall (security group)
asdf

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

5. Template is successfully created



Success
Successfully created my11(lt-008a87e86b9d52aa0).

▶ Actions log

Next Steps

Launch an instance

With On-Demand Instances, you pay for compute capacity by the second (for Linux, with a minimum of 60 seconds) or by the hour (for all other operating systems) with no long-term commitments or upfront payments. Launch an On-Demand Instance from your launch template.

Launch instance from this template

Create an Auto Scaling group from your template

Amazon EC2 Auto Scaling helps you maintain application availability and allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. You can use Auto Scaling to help ensure that you are running your desired number of Amazon EC2 instances during demand spikes to maintain performance and decrease capacity during lulls to reduce costs.

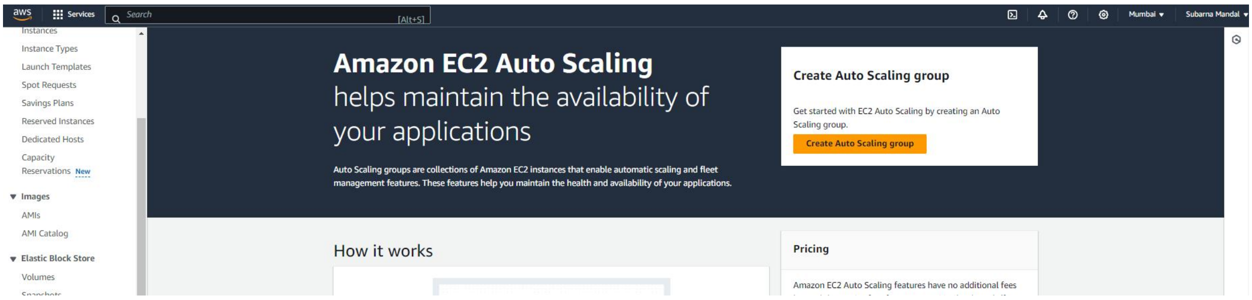
Create Auto Scaling group

Create Spot Fleet

A Spot Instance is an unused EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot Instance (of each instance type in each Availability Zone) is set by Amazon EC2, and adjusted gradually based on the long-term supply of and demand for Spot Instances. Spot Instances are well-suited for data-analysis, batch jobs, background processing, and optional tasks.

Create Spot Fleet

6. Once the launch template has been successfully created, navigate to the left pane and search for "Auto Scaling Groups". Then, select "Create Auto Scaling Group"



Amazon EC2 Auto Scaling helps maintain the availability of your applications

Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

Create Auto Scaling group

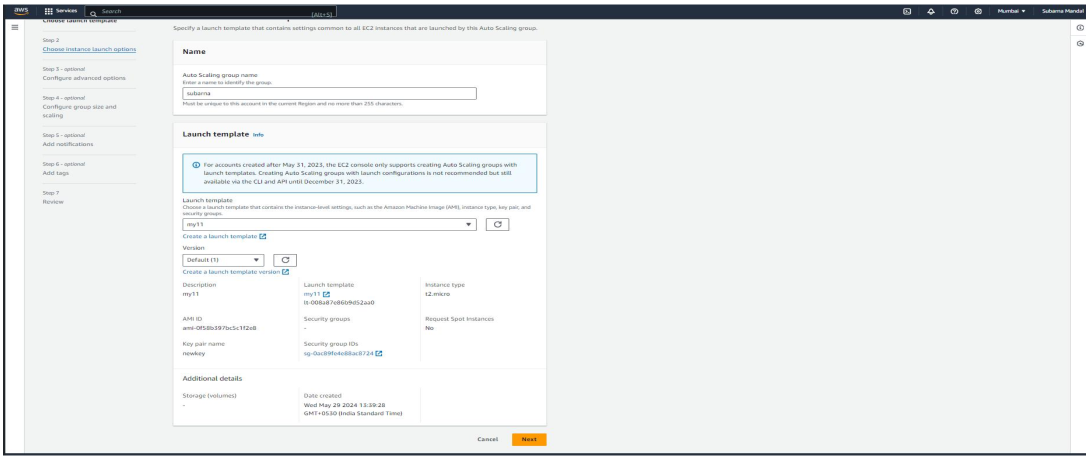
Get started with EC2 Auto Scaling by creating an Auto Scaling group.

How it works

Pricing

Amazon EC2 Auto Scaling features have no additional fees

7. Please specify a name for the scaling group . Select the template that was created in the preceding steps . Proceed to click on “Next”



Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name
Enter a name to identify the group.
subarna
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
my11
Create a launch template

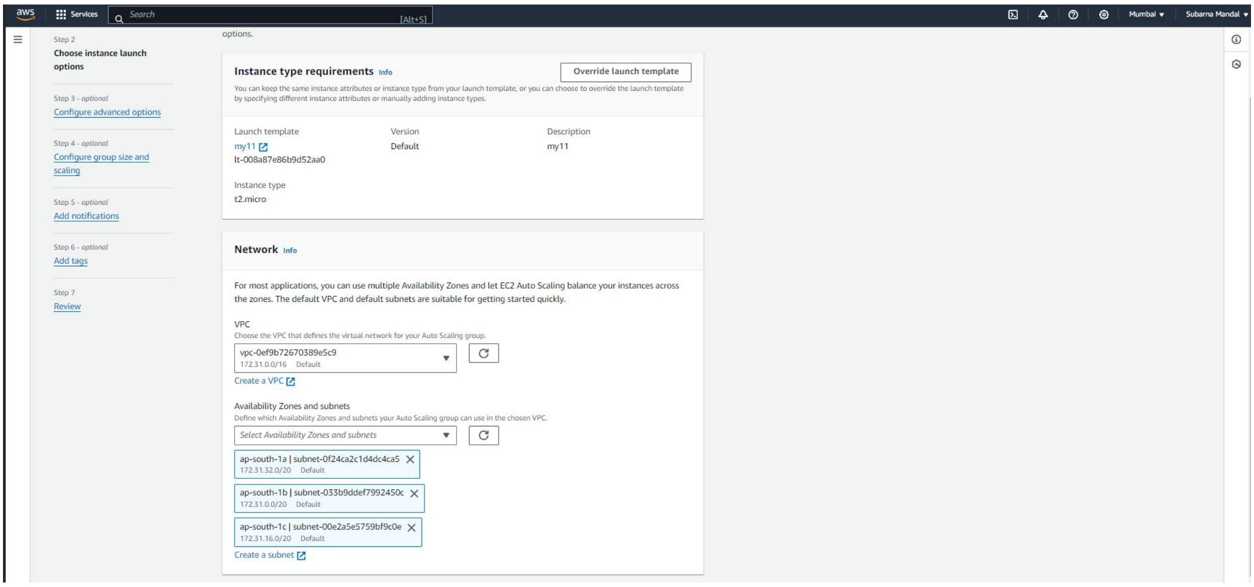
Version
Default (1)
Create a launch template version

Description	Launch template	Instance type
my11	my11 lt-008a87e86b9d52aa0	t2.micro
AMI ID	Security groups	Request Spot Instances
ami-0f58b397bc5c1f2e8	-	No
Key pair name	Security group IDs	
newkey	sg-0ac89fe4e88ac8724	

Additional details

Storage (volumes)	Date created
-	Wed May 29 2024 13:39:28 GMT+0530 (India Standard Time)

8. In the following step, choose all available availability zones and subnets, then proceed by clicking "Next"



options.

Instance type requirements Info

Override launch template

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

Launch template	Version	Description
my11 lt-008a87e86b9d52aa0	Default	my11

Instance type
t2.micro

Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-0ef9b72670389e5c9 ▾
172.31.0.0/16 Default
Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets ▾

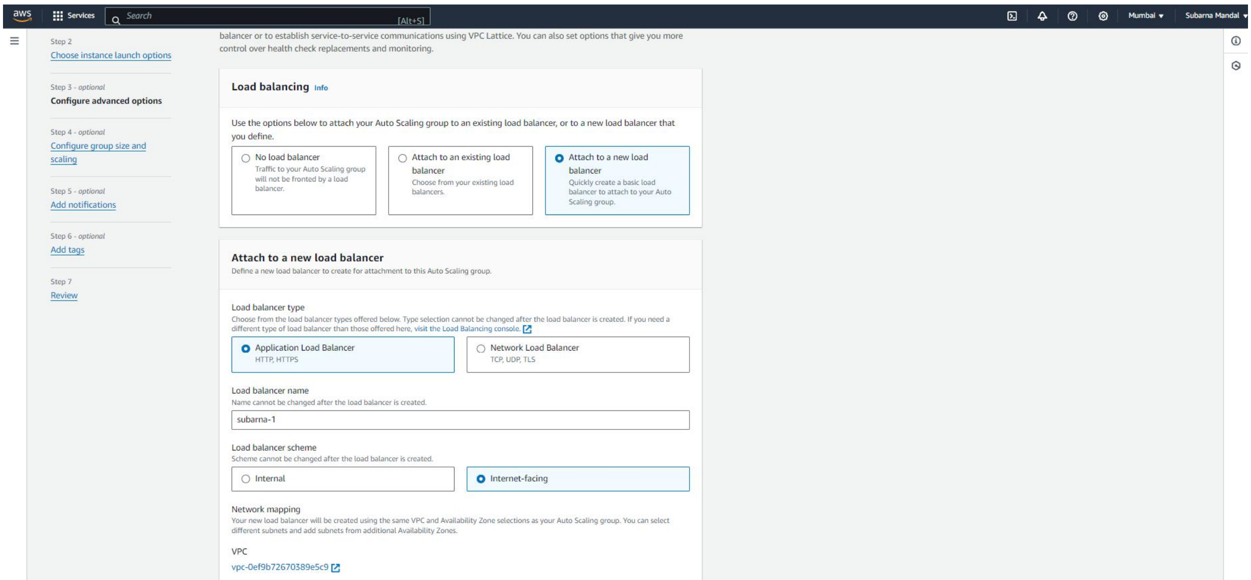
ap-south-1a | subnet-0f24ca2c1d4dc4ca5 ✕
172.31.32.0/20 Default

ap-south-1b | subnet-033b9ddef7992450c ✕
172.31.0.0/20 Default

ap-south-1c | subnet-00e2a5e5759bf9c0e ✕
172.31.16.0/20 Default

Create a subnet

9. In the subsequent step, begin by selecting "Attach to a new load balancer. Select "Application Load Balancer" as the load balancer type and "Internet-facing" as the load balancer scheme



balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

○ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

○ Attach to an existing load balancer
Choose from your existing load balancers.

● Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to a new load balancer
Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

● Application Load Balancer
HTTP, HTTPS

○ Network Load Balancer
TCP, UDP, TLS

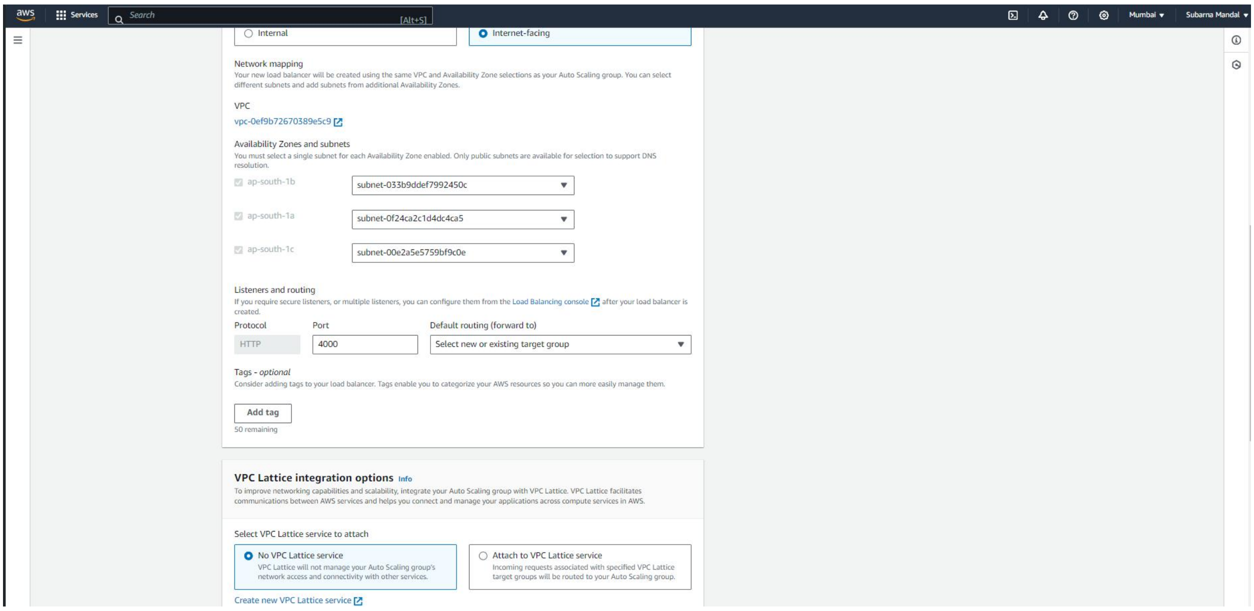
Load balancer name
Name cannot be changed after the load balancer is created.
subarna-1

Load balancer scheme
Scheme cannot be changed after the load balancer is created.
○ Internal
● Internet-facing

Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC
vpc-0ef9b72670389e5c9

10. Modify the HTTP port number from 80 to 4000 and designate the scaling group created for default routing.



○ Internal
● Internet-facing

Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC
vpc-0ef9b72670389e5c9

Availability Zones and subnets
You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

☑ ap-south-1b subnet-033b9ddef7992450c ▾
☑ ap-south-1a subnet-0f24ca2c1d4dc4ca5 ▾
☑ ap-south-1c subnet-00e2a5e5759bf9c0e ▾

Listeners and routing
If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	4000	Select new or existing target group ▾

Tags - *optional*
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add tag
50 remaining

VPC Lattice integration options Info
To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

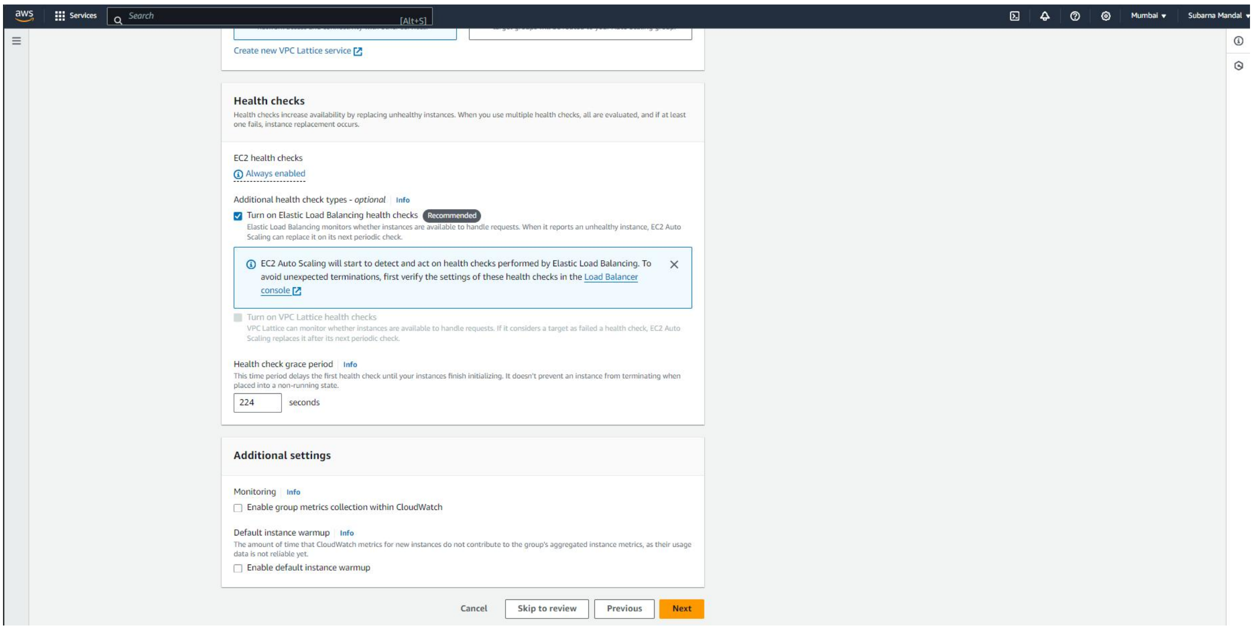
Select VPC Lattice service to attach

● No VPC Lattice service
VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

○ Attach to VPC Lattice service
Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

Create new VPC Lattice service

11. Enable the checkbox to activate health checks and specify a “health check grace period”, set here to 224 seconds. Without any further modifications, proceed to click on “Next”.



Create new VPC Lattice service

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks

Always enabled

Additional health check types - *optional* Info

☑ Turn on Elastic Load Balancing health checks Recommended

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

EC2 Auto Scaling will start to detect and act on health checks performed by Elastic Load Balancing. To avoid unexpected terminations, first verify the settings of these health checks in the Load Balancer console

□ Turn on VPC Lattice health checks

VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Health check grace period Info

This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

224 seconds

Additional settings

Monitoring Info

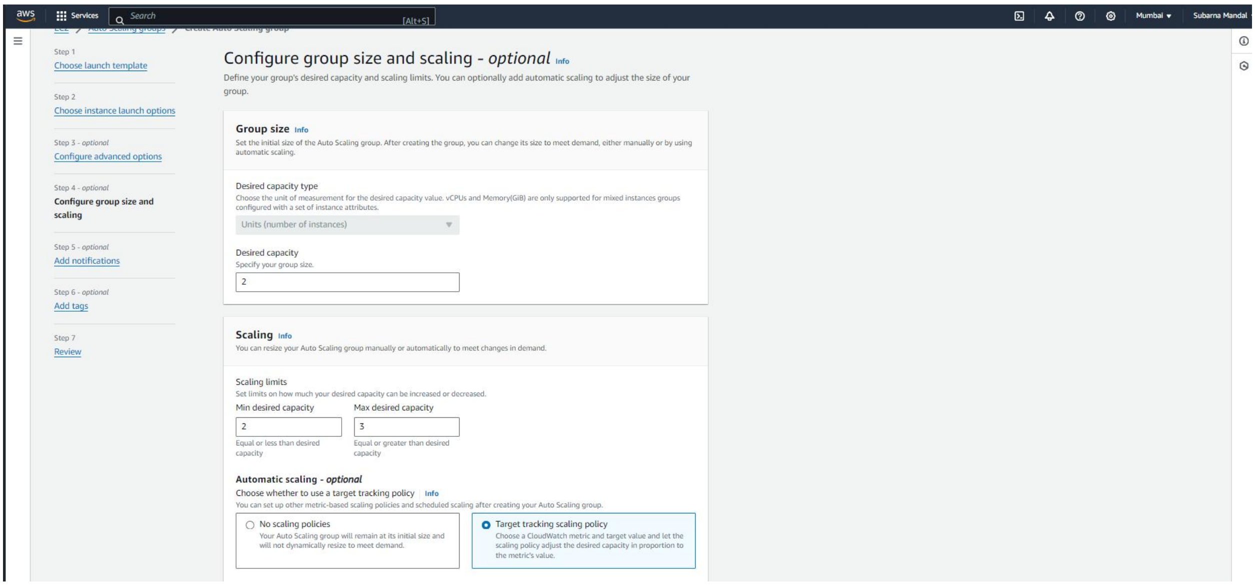
□ Enable group metrics collection within CloudWatch

Default instance warmup Info

The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.

□ Enable default instance warmup

12. In this step, specify the desired, minimum, and maximum capacities. Next, opt for the "Target Tracking Scaling Policy"



Step 1
Choose launch template

Step 2
Choose instance launch options

Step 3 - *optional*
Configure advanced options

Step 4 - *optional*
Configure group size and scaling

Step 5 - *optional*
Add notifications

Step 6 - *optional*
Add tags

Step 7
Review

Configure group size and scaling - *optional* Info

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity: 2 (Equal or less than desired capacity)

Max desired capacity: 3 (Equal or greater than desired capacity)

Automatic scaling - *optional*

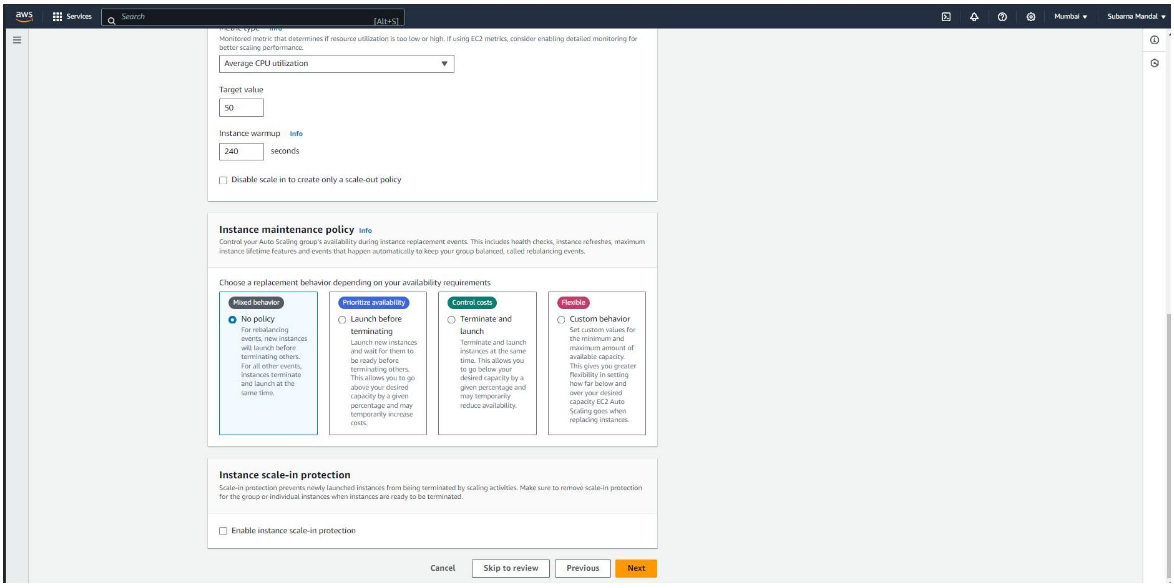
Choose whether to use a target tracking policy Info

You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

○ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

● Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

13. Configure the CPU utilization target value to 50. Additionally, set the instance warm-up time to 240 seconds. Proceed by clicking "Next" without making any changes



Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization

Target value

50

Instance warmup Info

240 seconds

□ Disable scale in to create only a scale-out policy

Instance maintenance policy Info

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Mixed behavior
● No policy
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

Prioritize availability
○ Launch before terminating
Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.

Control costs
○ Terminate and launch
Terminate and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may temporarily reduce availability.

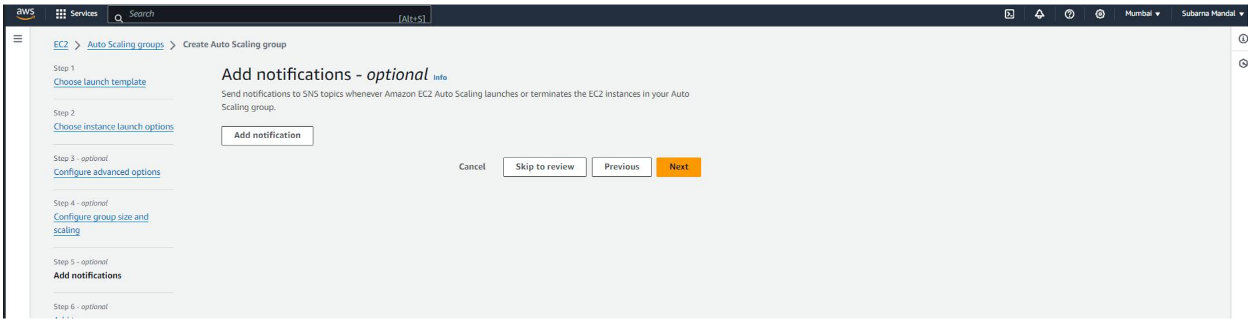
Flexible
○ Custom behavior
Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far below and over your desired capacity EC2 Auto Scaling goes when replacing instances.

Instance scale-in protection

Scale-in protection prevents newly launched instances from being terminated by scaling activities. Make sure to remove scale-in protection for the group or individual instances when instances are ready to be terminated.

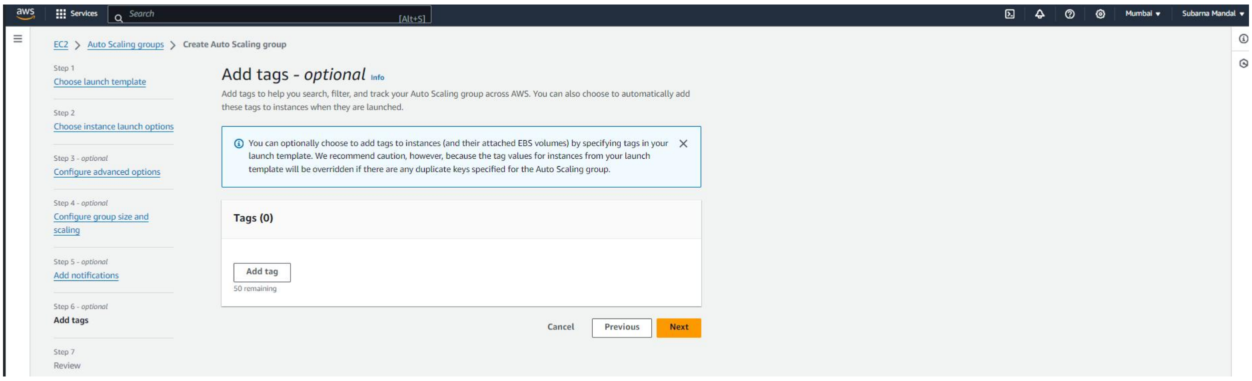
□ Enable instance scale-in protection

14. Click “Create Auto Scaling group”



Add notifications - *optional* Info

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.



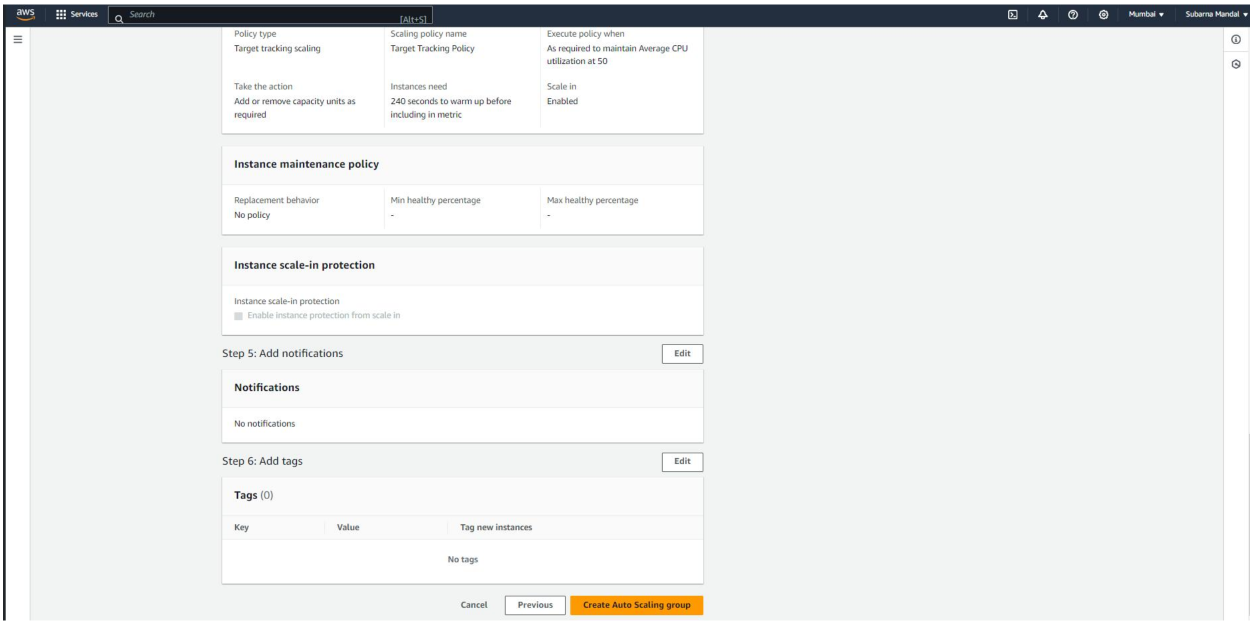
Add tags - *optional* Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to add tags to instances when they are launched.

You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group.

Tags (0)

50 remaining



Policy type: Target tracking scaling

Scaling policy name: Target Tracking Policy

Execute policy when: As required to maintain Average CPU utilization at 50

Take the action: Add or remove capacity units as required

Instances need: 240 seconds to warm up before including in metric

Scale in: Enabled

Instance maintenance policy

Replacement behavior: No policy

Min healthy percentage: -

Max healthy percentage: -

Instance scale-in protection

Instance scale-in protection

□ Enable instance protection from scale in

Step 5: Add notifications

Notifications

No notifications

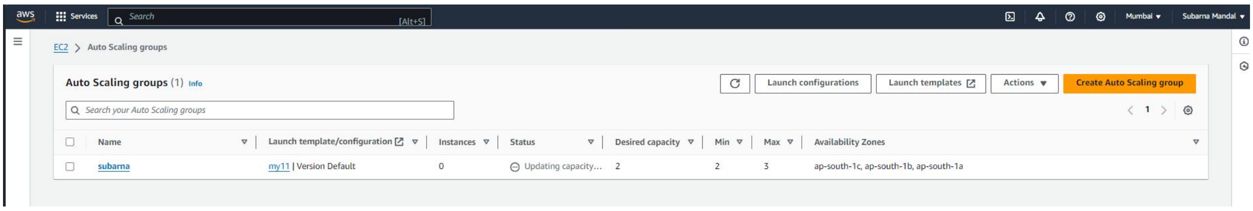
Step 6: Add tags

Tags (0)

Key	Value	Tag new instances

No tags

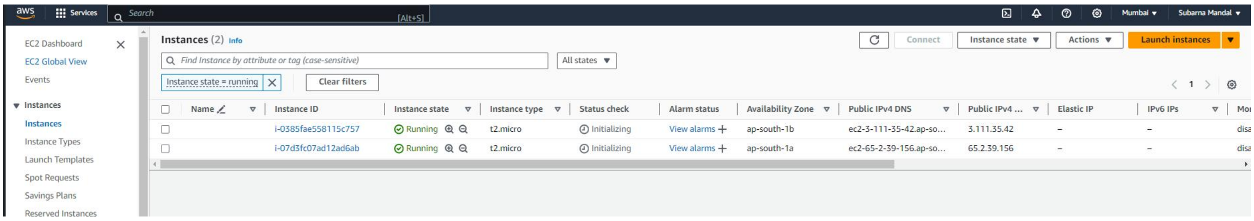
15. Auto scaling group created successfully



Auto Scaling groups (1) Info

□	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
□	subarna	my11 | Version Default	0	Updating capacity...	2	2	3	ap-south-1c, ap-south-1b, ap-south-1a

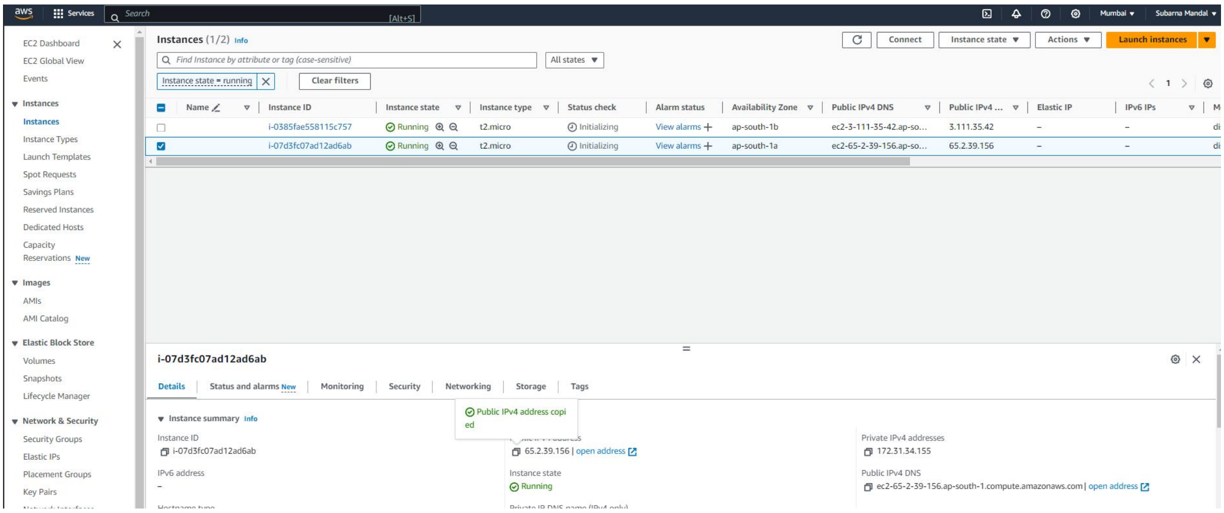
16. After the auto-scaling group is created, return to the EC2 dashboard and navigate to the "Instances" Since the capacity was given as 2, two instances are created. Now open any one of the instance by clicking on its id.



Instances (2) Info

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Mo
□		i-0385fae558115c757	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-3-111-35-42.ap-so...	3.111.35.42	–	–	disa
□		i-07d3fc07ad12ad6ab	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-65-2-39-156.ap-so...	65.2.39.156	–	–	disa

17. Choose any one of the instance IDs and copy the public IPv4 address.



Instances (1/2) Info

Find Instance by attribute or tag (case-sensitive)

Instance state = running

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
□		i-0385fae558115c757	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-3-111-35-42.ap-so...	3.111.35.42	–	–
☑		i-07d3fc07ad12ad6ab	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-65-2-39-156.ap-so...	65.2.39.156	–	–

i-07d3fc07ad12ad6ab

Details | Status and alarms New | Monitoring | Security | Networking | Storage | Tags

Public IPv4 address copied

Instance summary Info

Instance ID
i-07d3fc07ad12ad6ab

65.2.39.156 | open address

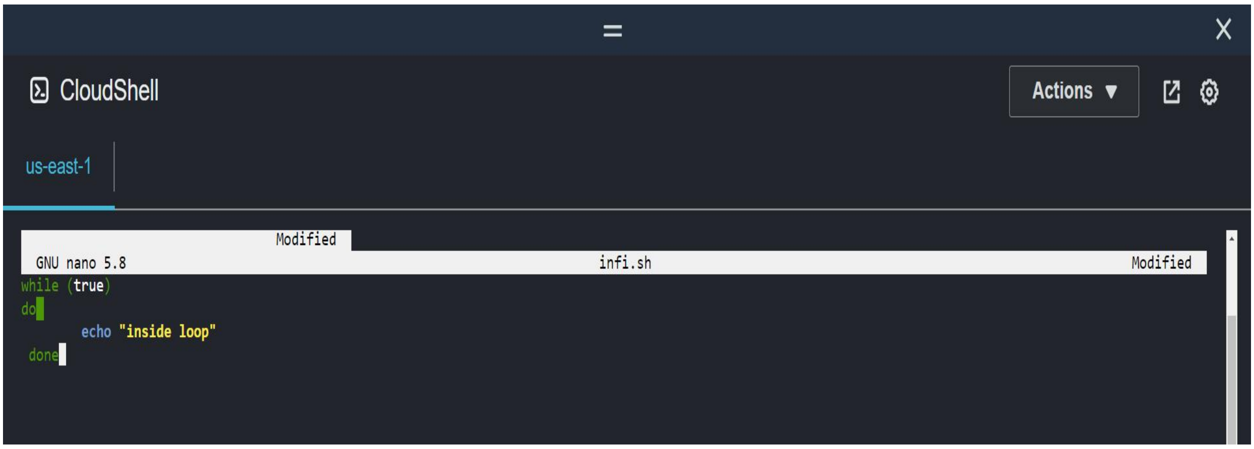
Private IPv4 addresses
172.31.34.155

IPv6 address
–

Instance state
Running

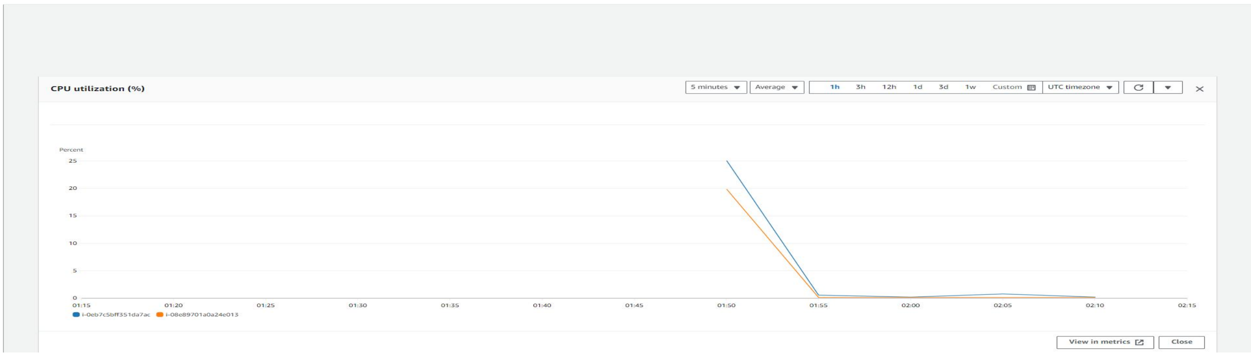
Public IPv4 DNS
ec2-65-2-39-156.ap-south-1.compute.amazonaws.com | open address

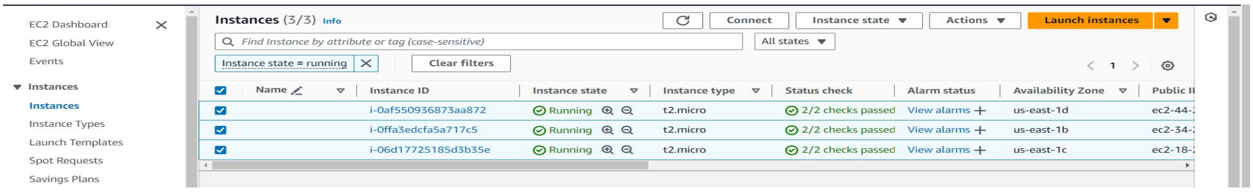
18. Open Bitwise SSH Client and log in using the IPv4 address we copied earlier. If bitvise not open then open CloudShell terminal and write "**sudo nano infi.sh**" command so creates a .sh file and then write this following code in the file "infi.sh" to run an infinite loop. Press CTRL+X ,Y then enter for save the file.
write "**sudo chmod 777 infi.sh**" (to provide all permission to the file)
sh infi.sh run the .sh file infinite time



```
                     Modified
 GNU nano 5.8                          infi.sh                              Modified
while (true)
do
        echo "inside loop"
 done
```

19. Return to AWS and select both running instances. Below, locate the monitoring options, and choose "CPU utilization". Then, enlarge the view.
20. From the panel above, select "Local timezone." The graph displays CPU utilization for both instances. When the CPU utilization exceeds the limit for both instances, another instance is created, as we have set the maximum capacity to 3
CPU utilization of two instance





Instances (3/3) Info

Find Instance by attribute or tag (case-sensitive)

Instance state = running

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public I
☑		i-0af550936873aa872	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1d	ec2-44-
☑		i-0ffa3edcfa5a717c5	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-34-
☑		i-06d17725185d3b35e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c	ec2-18-

After exceed CPU utilization limit the three instances are created