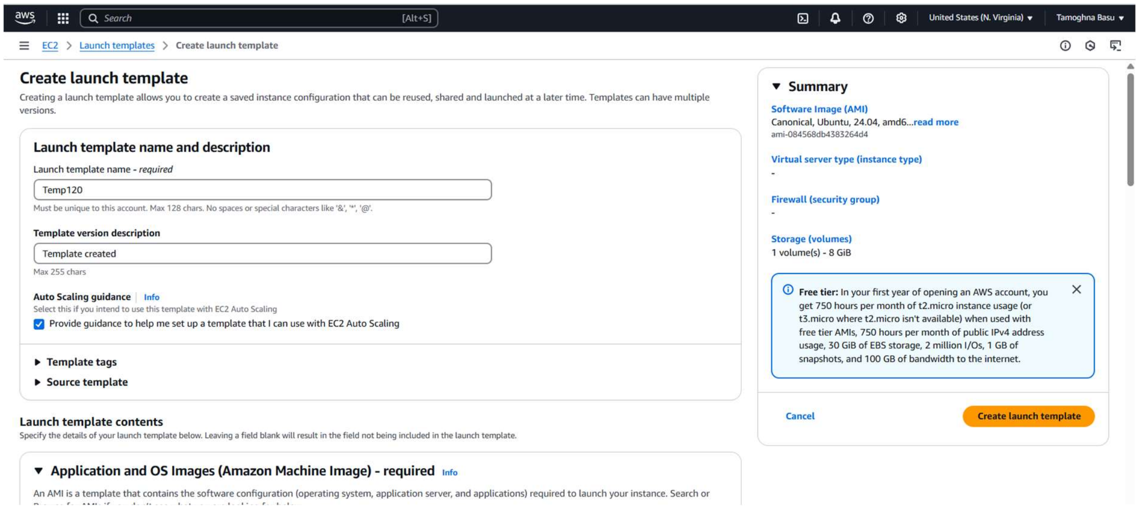
Assignment-11

Problem Definition: Build scaling plans in AWS that balance the load on different EC2 instance.

Step-1: Signing in to the AWS Management console,go to lauch template.A template name should be given along with a description and the auto-scaling option is to be checked.Then ubuntu is selected from quickstart.


aws
Search [Alt+S]
United States (N. Virginia) ▾
Tamoghna Basu ▾
EC2 > Launch templates > Create launch template

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*

Temp120

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

Template version description

Template created

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

▾ Summary

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd6...read more

ami-084568db4383264d4

Virtual server type (instance type)

-

Firewall (security group)

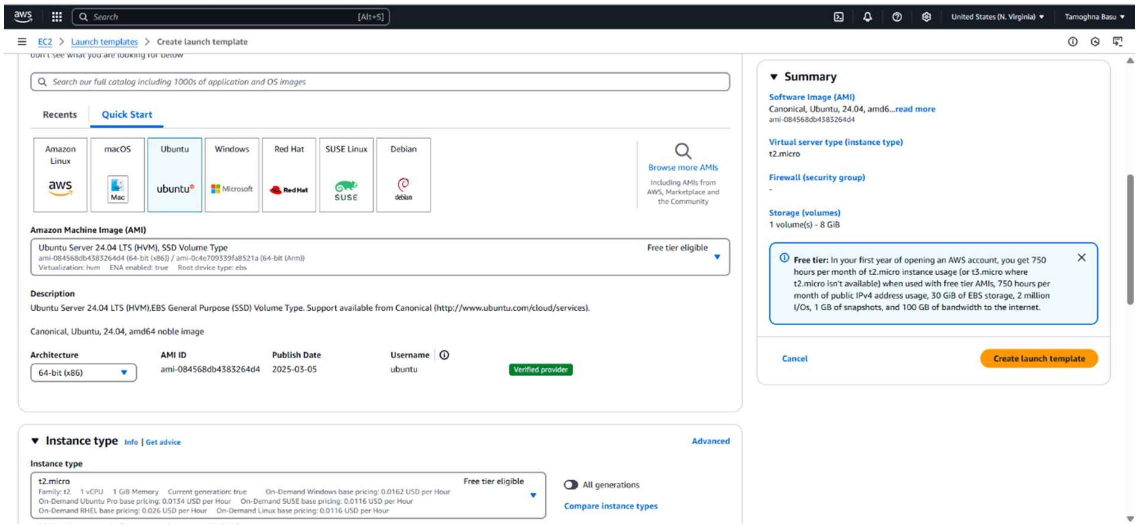
-

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Step-2: Instance type is selected as t2.micro.



aws Search [Alt+S] United States (N. Virginia) ▼ Tamoghna Basu ▼

EC2 > Launch templates > Create launch template

Don't see what you are looking for below

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

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Linux | Debian

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)
Ubuntu Server 24.04 LTS (HVM), SSD Volume Type — Free tier eligible
ami-084568db4383264d4 (64-bit (x86)) / ami-0c4e709339fa8521a (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture	AMI ID	Publish Date	Username	
64-bit (x86)	ami-084568db4383264d4	2025-03-05	ubuntu	Verified provider

▼ **Instance type** Info | Get advice — Advanced

Instance type
t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour On-Demand Linux base pricing: 0.0116 USD per Hour

All generations
Compare instance types

▼ **Summary**
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-084568db4383264d4

Virtual server type (instance type)
t2.micro

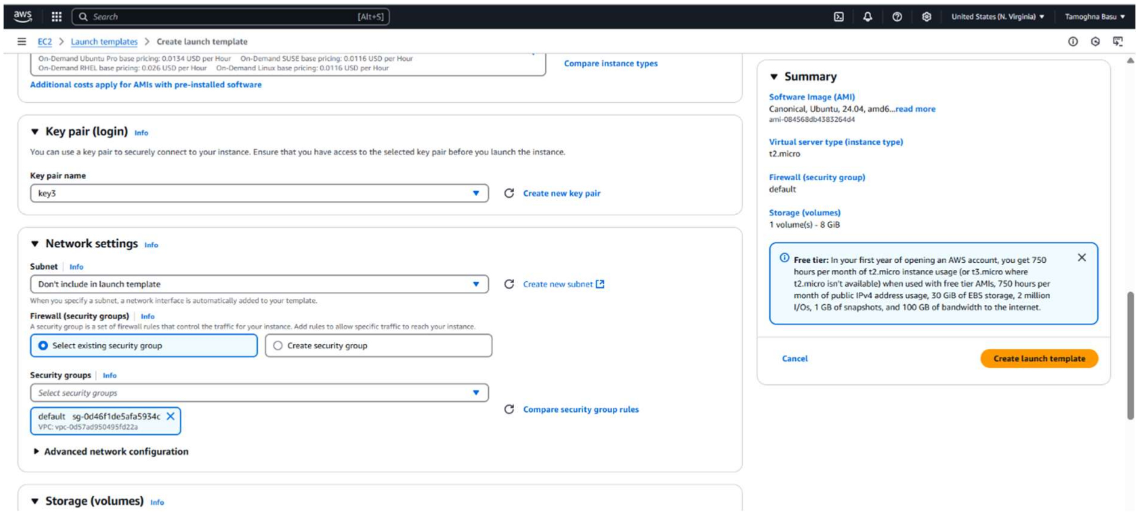
Firewall (security group)
-

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

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Create launch template

Step-3:A key-pair is selected along with the secuirity group and the user data is given.Create launch template is selected to create a new template.



aws Search [Alt+S] United States (N. Virginia) ▼ Tamoghna Basu ▼

EC2 > Launch templates > Create launch template

On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour On-Demand Linux base pricing: 0.0116 USD per Hour
Compare instance types
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
key3
Create new key pair

▼ **Network settings** Info

Subnet | Info
Don't include in launch template
Create new subnet
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
default sg-0d46f1de5afa5934c
VPC: vpc-0d57ad950495fd22a
Compare security group rules

▸ Advanced network configuration

▼ **Storage (volumes)** Info

▼ **Summary**
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-084568db4383264d4

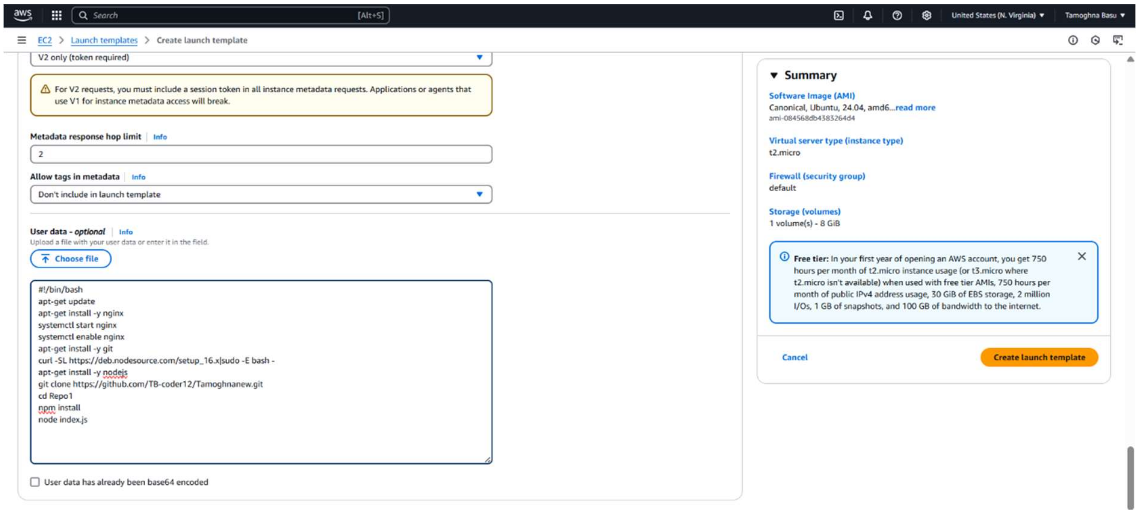
Virtual server type (instance type)
t2.micro

Firewall (security group)
default

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

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Create launch template



aws
Search [Alt+S]
United States (N. Virginia) ▼
Tamoghna Basu ▼
EC2 > Launch templates > Create launch template

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.

```
#!/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 https://github.com/TB-coder12/Tamoghnanew.git
cd Repo1
npm install
node index.js
```

☐ User data has already been base64 encoded

▼ Summary

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-084568db4383264d4

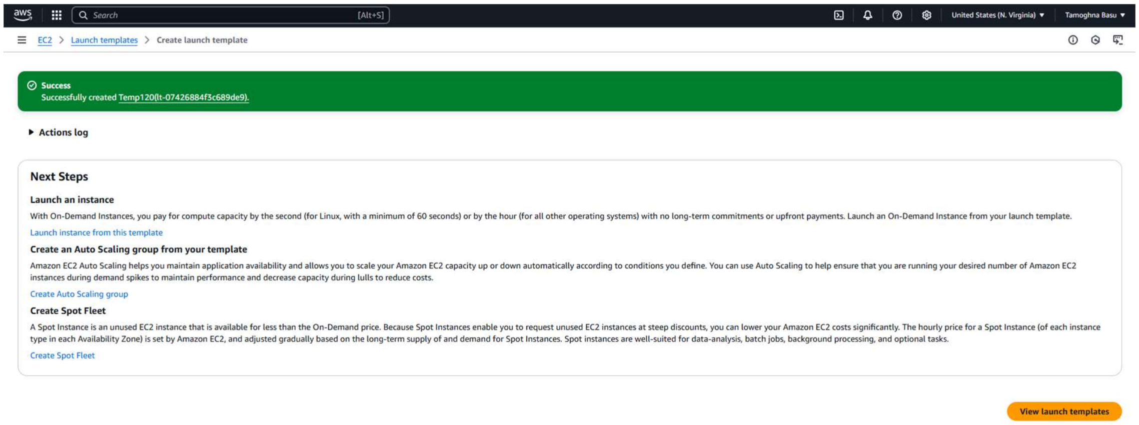
Virtual server type (instance type)
t2.micro

Firewall (security group)
default

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

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel
Create launch template



aws
Search [Alt+S]
United States (N. Virginia) ▼
Tamoghna Basu ▼
EC2 > Launch templates > Create launch template

Success
Successfully created Temp120(lt-07426884f3c689de9).

▸ 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.

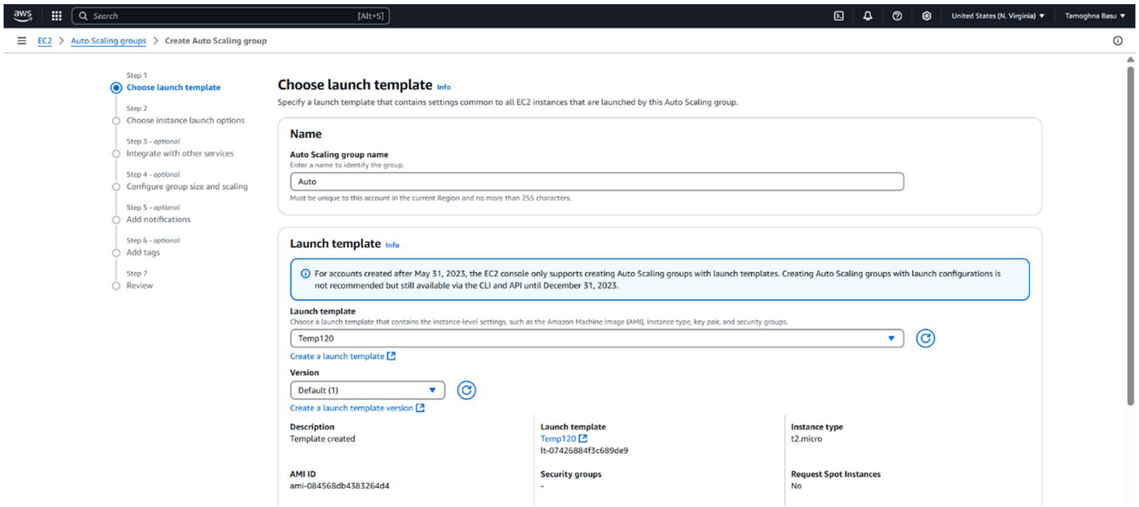
Create Auto Scaling group

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

Step-4:Auto-scaling group is reached ,a name is given,template is selected,zones are selected,along with Balanced best effort.


aws
Search [Alt+S]
United States (N. Virginia) ▼
Tamoghna Basu ▼
EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1
● Choose launch template

Step 2
○ Choose instance launch options

Step 3 - *optional*
○ Integrate with other services

Step 4 - *optional*
○ Configure group size and scaling

Step 5 - *optional*
○ Add notifications

Step 6 - *optional*
○ Add tags

Step 7
○ Review

Choose launch template Info

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.
Auto
Must be unique to this account in the current Region and no more than 255 characters.

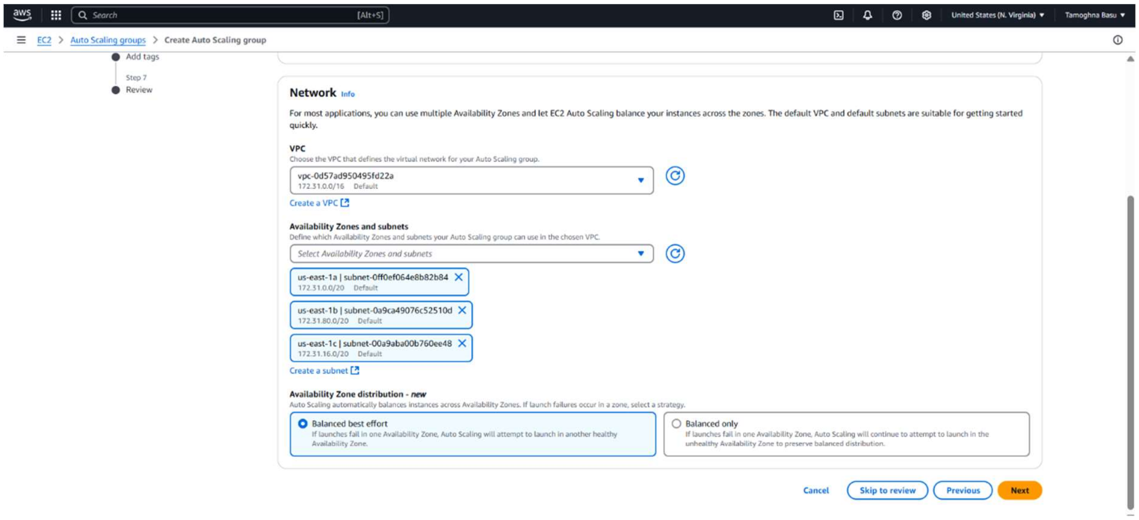
Launch template Info

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.
Temp120 ▼
Create a launch template

Version
Default (1) ▼
Create a launch template version

Description Template created	**Launch template** Temp120 lt-07426884f3c689de9	**Instance type** t2.micro
AMI ID ami-084568db4383264d4	**Security groups** -	**Request Spot Instances** No


aws
Search [Alt+S]
United States (N. Virginia) ▼
Tamoghna Basu ▼
EC2 > Auto Scaling groups > Create Auto Scaling group

● Add tags

Step 7
● Review

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-0d57ad950495fd22a
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 ▼

us-east-1a | subnet-0ff0ef064e8b82b84 ✕
172.31.0.0/20 Default

us-east-1b | subnet-0a9ca49076c52510d ✕
172.31.80.0/20 Default

us-east-1c | subnet-00a9aba00b760ee48 ✕
172.31.16.0/20 Default

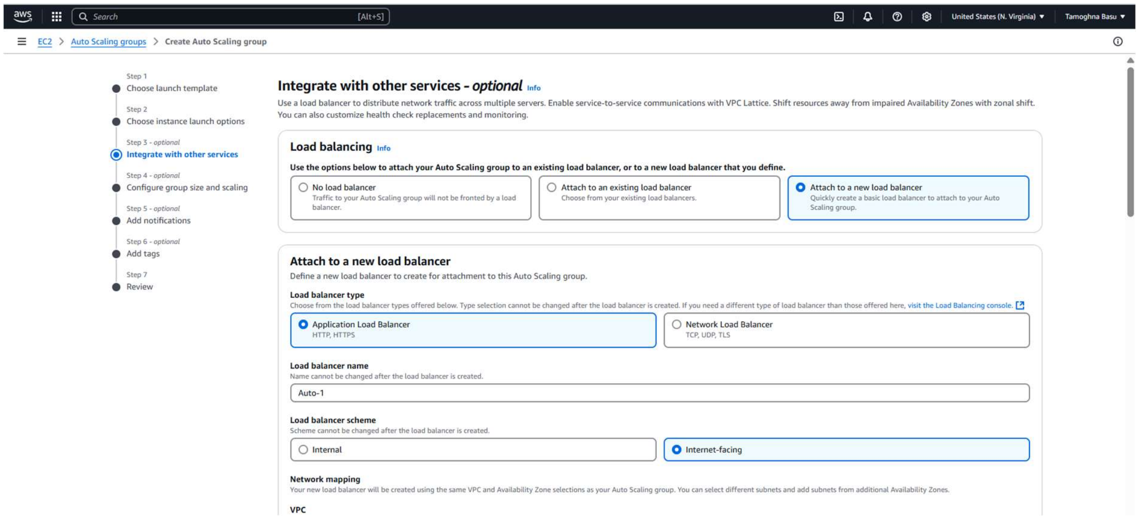
Create a subnet

Availability Zone distribution - ***new***
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

● Balanced best effort
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

○ Balanced only
If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

Step-5:On the next page,attach to a new load balancer is selected along with Internet-facing as the load balancer scheme.Additional health check types is enabled.


aws
Search
[Alt+S]
United States (N. Virginia) ▾
Tamoghna Basu ▾
EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review

Integrate with other services - *optional* Info

Use a load balancer to distribute network traffic across multiple servers. Enable service-to-service communications with VPC Lattice. Shift resources away from impaired Availability Zones with zonal shift. You can also customize 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.
Auto-1

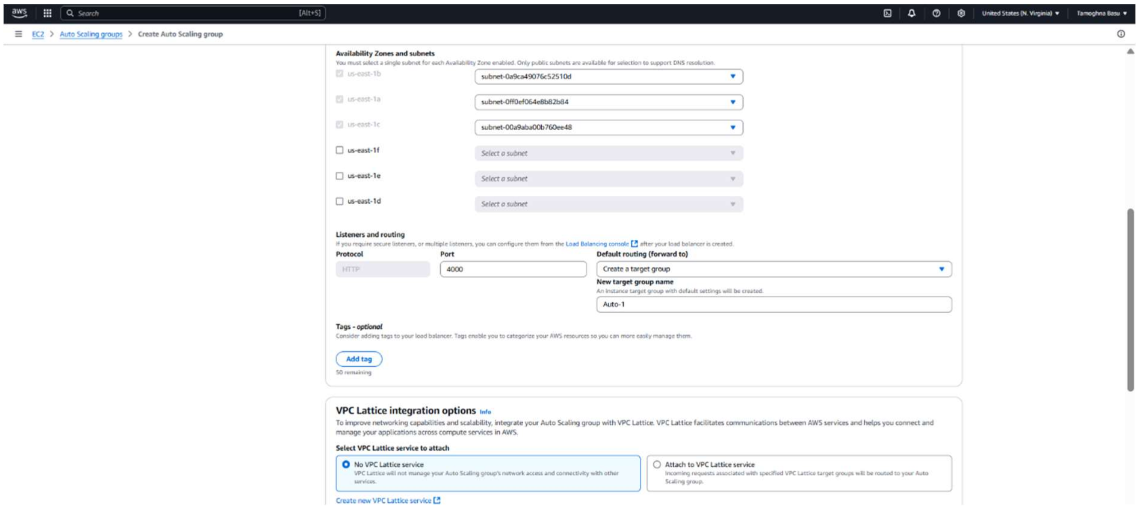
Load balancer scheme
Scheme cannot be changed after the load balancer is created.

○ 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


aws
Search
[Alt+S]
United States (N. Virginia) ▾
Tamoghna Basu ▾
EC2 > Auto Scaling groups > Create Auto Scaling group

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.

☑ us-east-1b subnet-0a9ca49076c52510d
☑ us-east-1a subnet-0ff0ef064e8b82b84
☑ us-east-1c subnet-00a9aba00b760ee48
□ us-east-1f Select a subnet
□ us-east-1e Select a subnet
□ us-east-1d Select a subnet

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
HTTP

Port
4000

Default routing (forward to)
Create a target group

New target group name
An instance target group with default settings will be created.
Auto-1

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

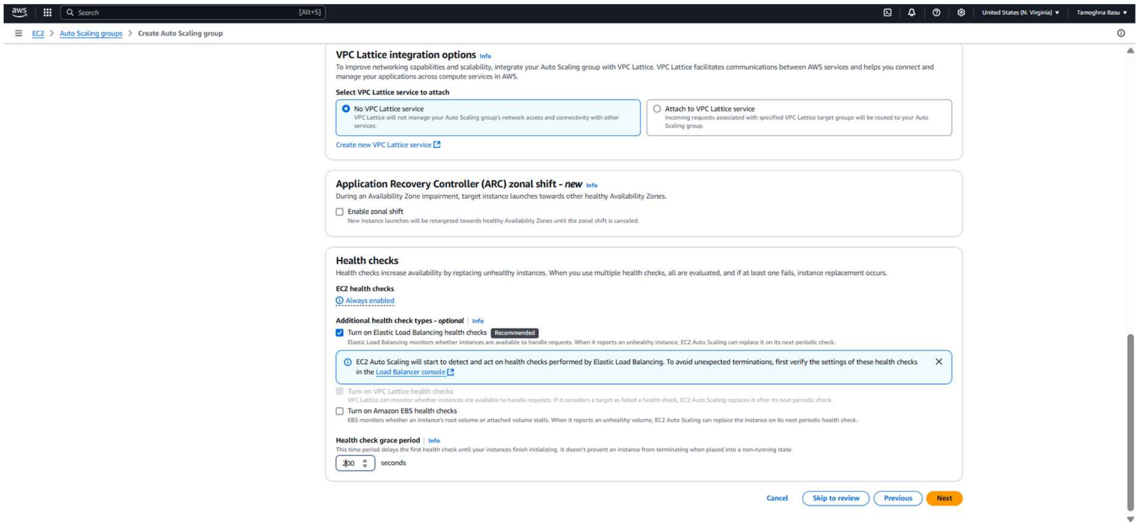
To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

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


aws
Search
[Alt+S]
United States (N. Virginia) ▾
Tamoghna Basu ▾
EC2 > Auto Scaling groups > Create Auto Scaling group

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

Application Recovery Controller (ARC) zonal shift - *new* Info

During an Availability Zone impairment, target instance launches towards other healthy Availability Zones.

□ Enable zonal shift
New instance launches will be retargeted towards healthy Availability Zones until the zonal shift is canceled.

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.

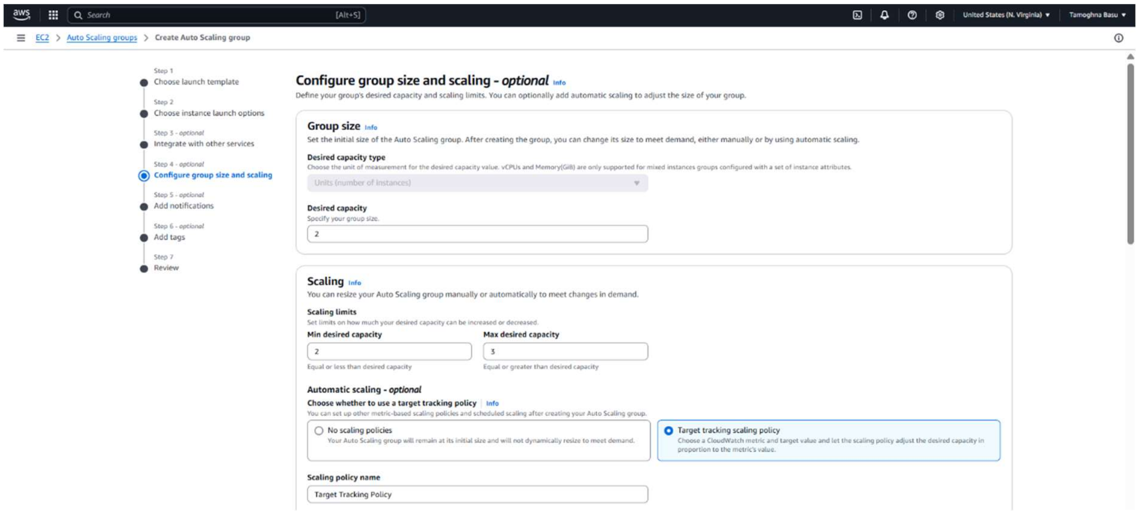
EC2 Auto Scaling will start to detect and act on health checks performed by Elastic Load Balancing. To avoid unexpected terminations, first verify the settings of these health checks in the Load Balancer console

□ 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.

□ Turn on Amazon EBS health checks
EBS monitors whether an instance's root volume or attached volume stalls. When it reports an unhealthy volume, EC2 Auto Scaling can replace the instance on its next periodic health 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.
200 seconds

Step-6:In configure group and scaling,desired capacity is selected as 2,along with min value as 2 and max as 3.Target tracking scaling policy is selected.Instance warmup is set for 200 sec.



Configure group size and scaling - *optional*

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

Group size

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

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

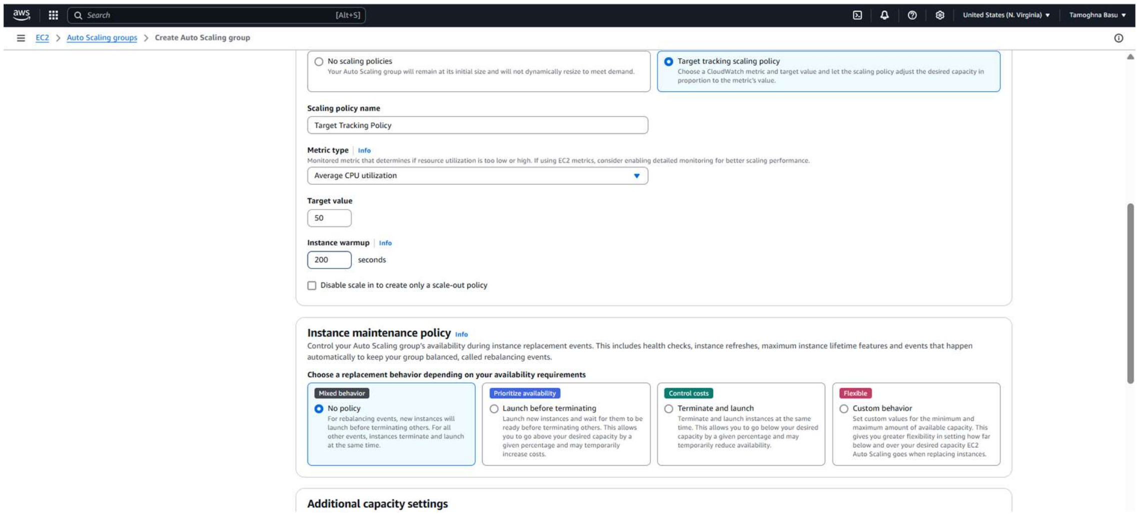
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

○ 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.

Scaling policy name

Target Tracking Policy


aws
Search
[Alt+S]
United States (N. Virginia) ▾
Tamoghna Basu ▾
EC2 > Auto Scaling groups > Create Auto Scaling group
Info

○ 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.

Scaling policy name

Target Tracking Policy

Metric type

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

200 seconds

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

Instance maintenance policy

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

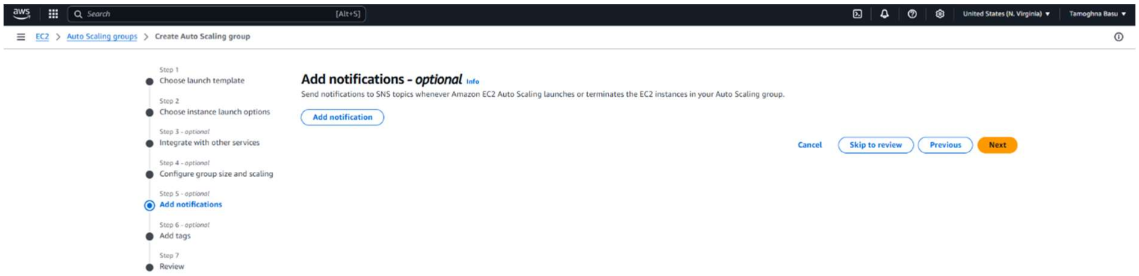
Mixed behavior
● No policy
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

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.

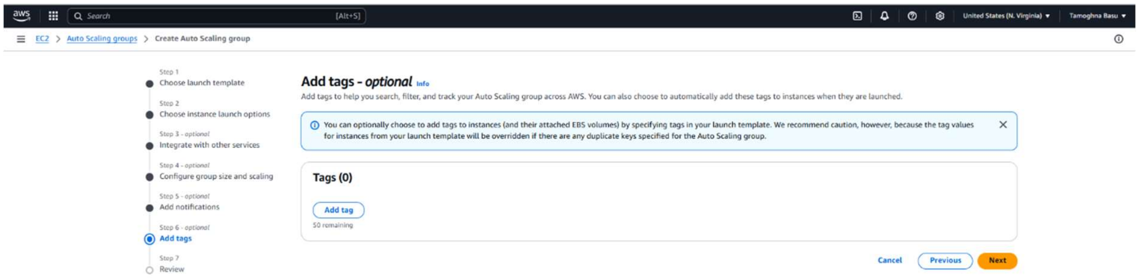
Flexible
○ Custom behavior
Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far below and over your desired capacity EC2 Auto Scaling goes when replacing instances.

Additional capacity settings



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.



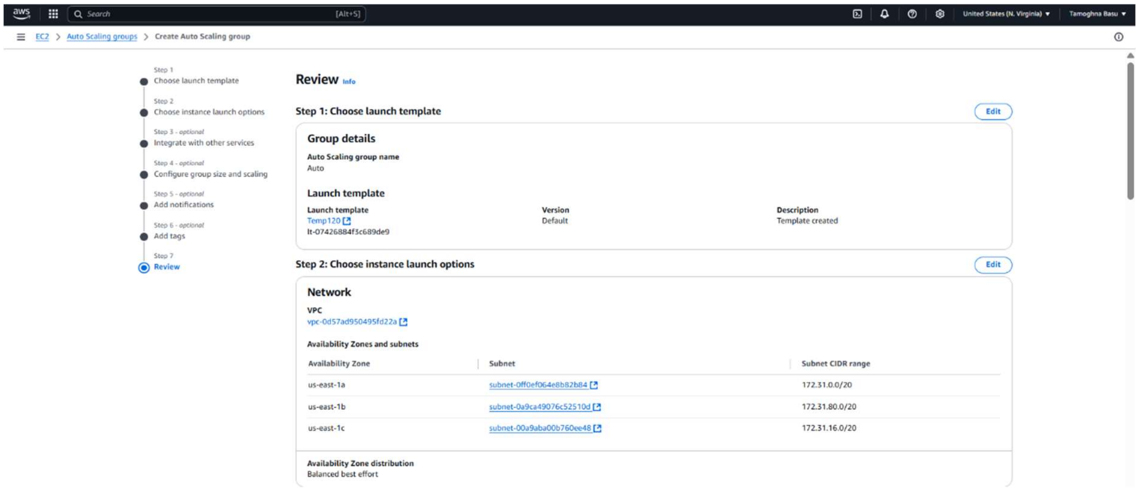
Add tags - *optional* Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these 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



Review Info

Step 1: Choose launch template

Group details

Auto Scaling group name
Auto

Launch template

Launch template	Version	Description
Temp120 lt-07426884f3c689de9	Default	Template created

Step 2: Choose instance launch options

Network

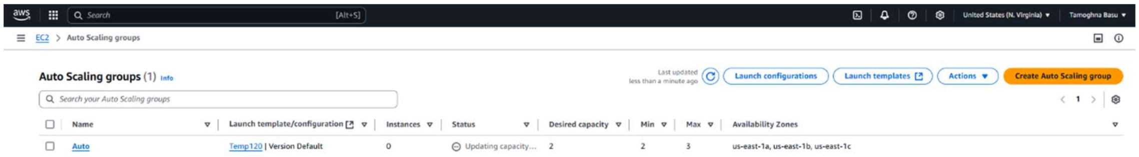
VPC
vpc-0d57ad950495fd22a

Availability Zones and subnets

Availability Zone	Subnet	Subnet CIDR range
us-east-1a	subnet-0ff0ef064e8b82b84	172.31.0.0/20
us-east-1b	subnet-0a9ca49076c52510d	172.31.80.0/20
us-east-1c	subnet-00a9aba00b760ee48	172.31.16.0/20

Availability Zone distribution
Balanced best effort

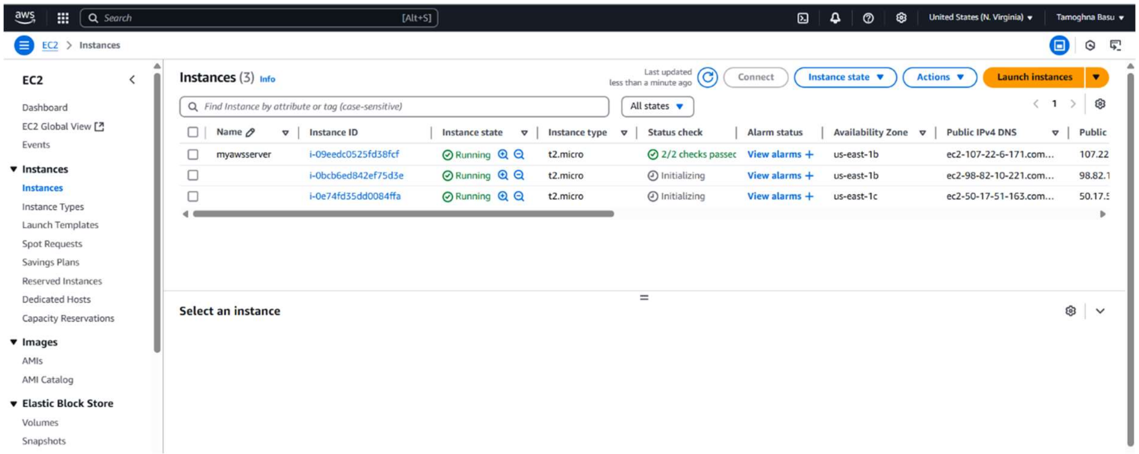
Step-7:Without making further modifications,simply clicking on next ,finally the auto scaling group has been created.



aws
Search [Alt+S]
United States (N. Virginia)
Tamoghna Basu
EC2 > Auto Scaling groups
Last updated less than a minute ago
Launch configurations
Launch templates
Actions
Create Auto Scaling group
Search your Auto Scaling groups

Auto Scaling groups (1) Info

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
Auto	Temp120 | Version Default	0	Updating capacity...	2	2	3	us-east-1a, us-east-1b, us-east-1c

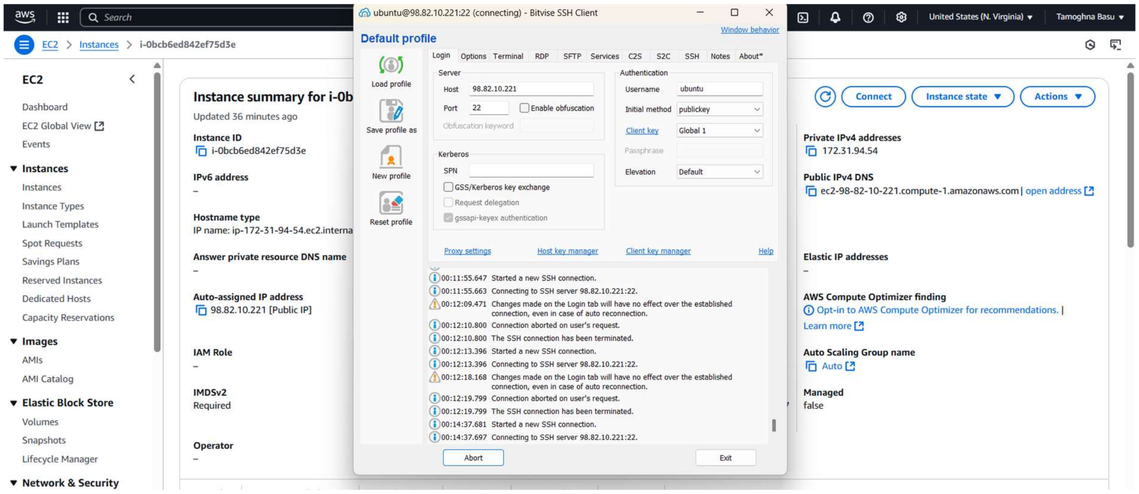


Instances (3) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public
myawsserver	i-09eedc0525fd38fcf	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1b	ec2-107-22-6-171.com...	107.22
	i-0bcb6ed842ef75d3e	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2-98-82-10-221.com...	98.82.1
	i-0e74fd35dd0084ffa	Running	t2.micro	Initializing	View alarms +	us-east-1c	ec2-50-17-51-163.com...	50.17.5

Select an instance

Step-8:Bitvise is opened,properly logged in with the key-pair and in the terminal following commands are written.



Instance summary for i-0b

Updated 36 minutes ago

Instance ID
i-0bcb6ed842ef75d3e

IPv6 address
–

Hostname type
IP name: ip-172-31-94-54.ec2.interna

Answer private resource DNS name
–

Auto-assigned IP address
98.82.10.221 [Public IP]

IAM Role
–

IMDSv2
Required

Operator
–

Private IPv4 addresses
172.31.94.54

Public IPv4 DNS
ec2-98-82-10-221.compute-1.amazonaws.com | open address

Elastic IP addresses
–

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. |
Learn more

Auto Scaling Group name
Auto

Managed
false

ubuntu@98.82.10.221:22 (connecting) - Bitvise SSH Client

Default profile

Server
Host: 98.82.10.221
Port: 22 ☐ Enable obfuscation
Obfuscation keyword

Kerberos
SPN
☐ GSS/Kerberos key exchange
☐ Request delegation
☑ gssapi-keyex authentication

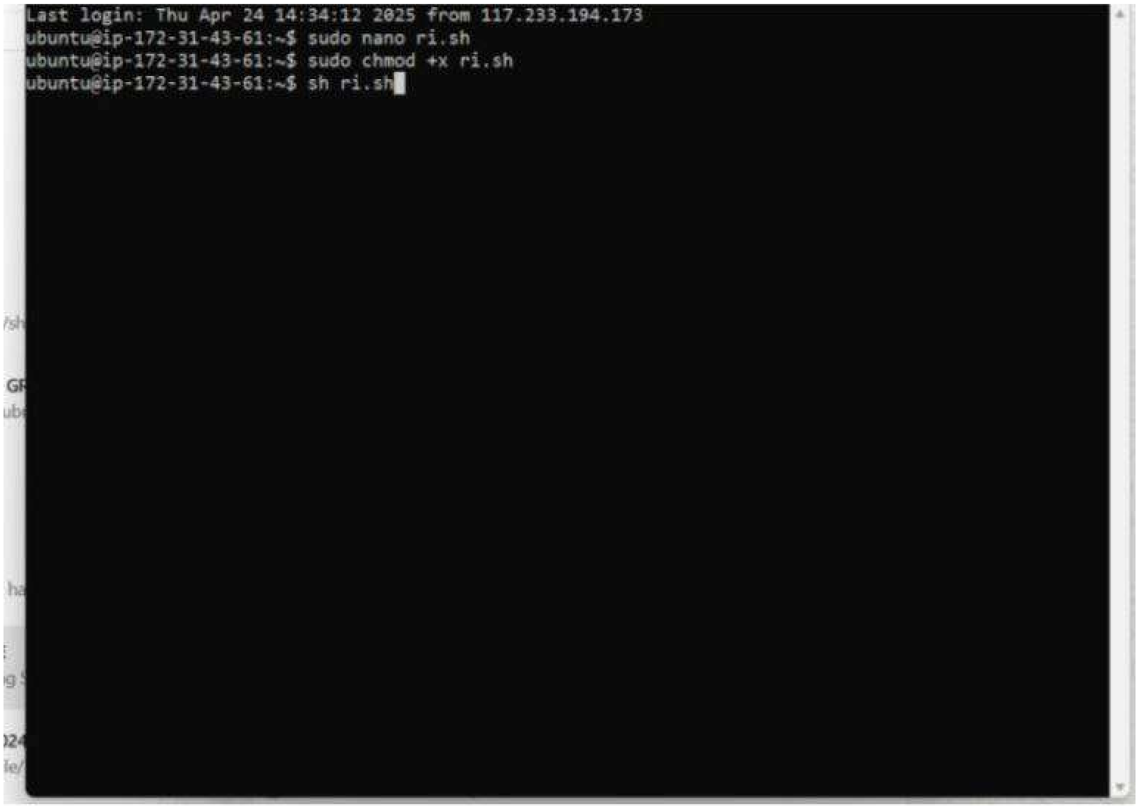
Authentication
Username: ubuntu
Initial method: publickey
Client key: Global 1
Passphrase
Elevation: Default

```
00:11:55.647 Started a new SSH connection.
00:11:55.663 Connecting to SSH server 98.82.10.221:22.
00:12:09.471 Changes made on the Login tab will have no effect over the established
             connection, even in case of auto reconnection.
00:12:10.800 Connection aborted on user's request.
00:12:10.800 The SSH connection has been terminated.
00:12:13.396 Started a new SSH connection.
00:12:13.396 Connecting to SSH server 98.82.10.221:22.
00:12:18.168 Changes made on the Login tab will have no effect over the established
             connection, even in case of auto reconnection.
00:12:19.799 Connection aborted on user's request.
00:12:19.799 The SSH connection has been terminated.
00:14:37.681 Started a new SSH connection.
00:14:37.697 Connecting to SSH server 98.82.10.221:22.
```

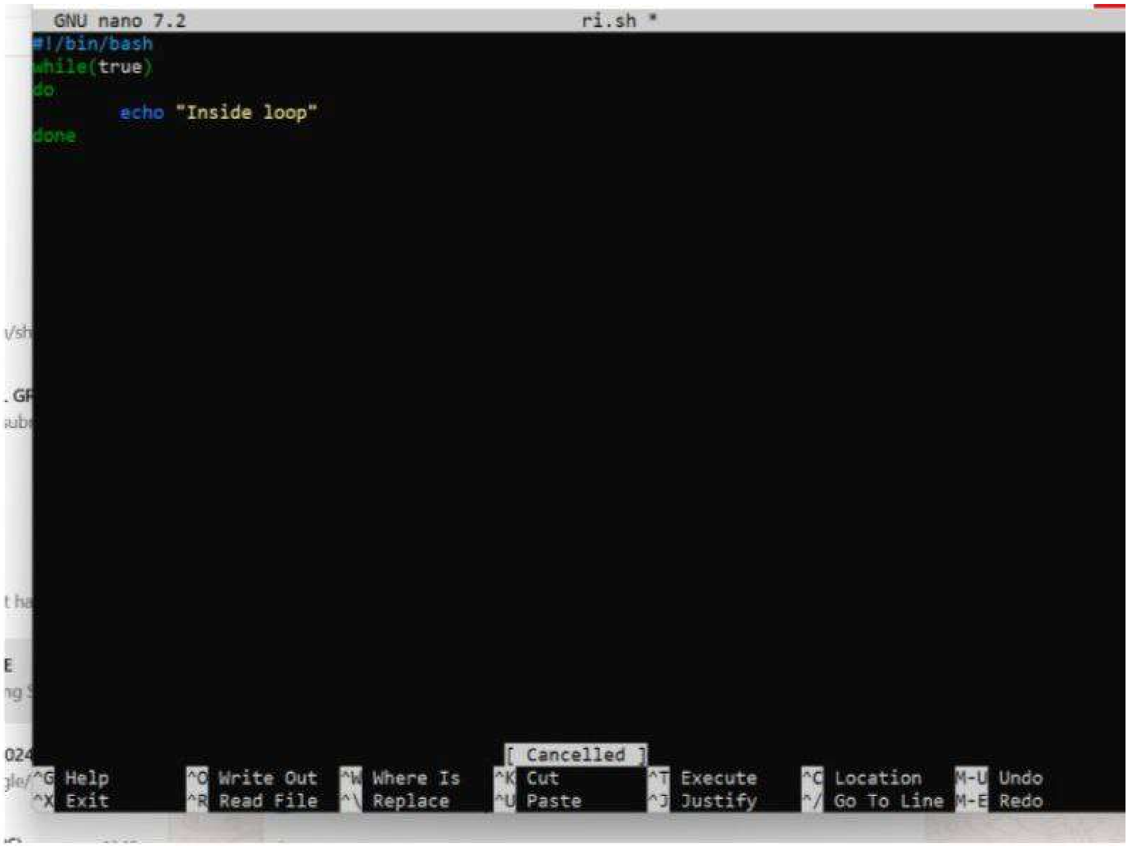
//Sudo nano ri.sh

Inside the file we have to write

```
#!/bin/bash
while(true)
do
    echo "Inside loop"
done
//sudo chmod +x ri.sh
//sh ri.sh
```

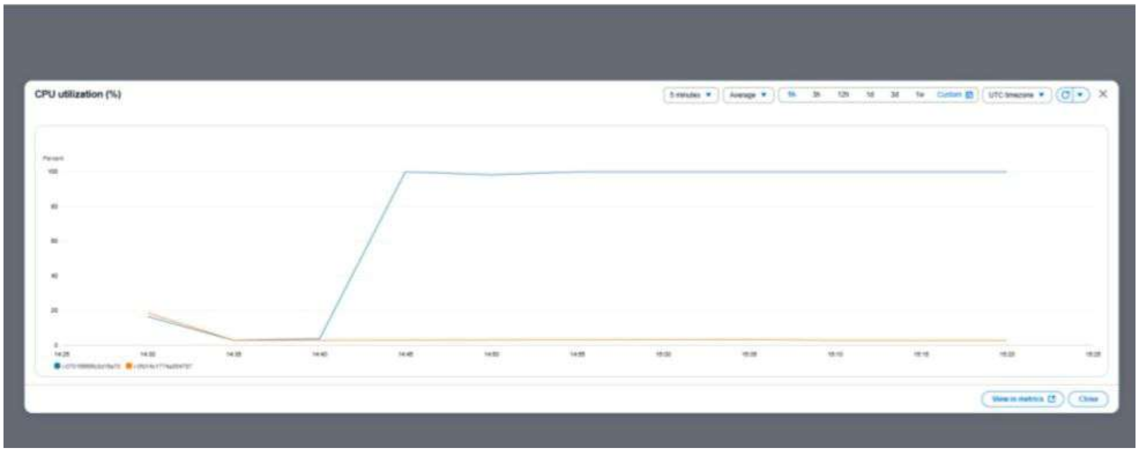


```
Last login: Thu Apr 24 14:34:12 2025 from 117.233.194.173
ubuntu@ip-172-31-43-61:~$ sudo nano ri.sh
ubuntu@ip-172-31-43-61:~$ sudo chmod +x ri.sh
ubuntu@ip-172-31-43-61:~$ sh ri.sh
```

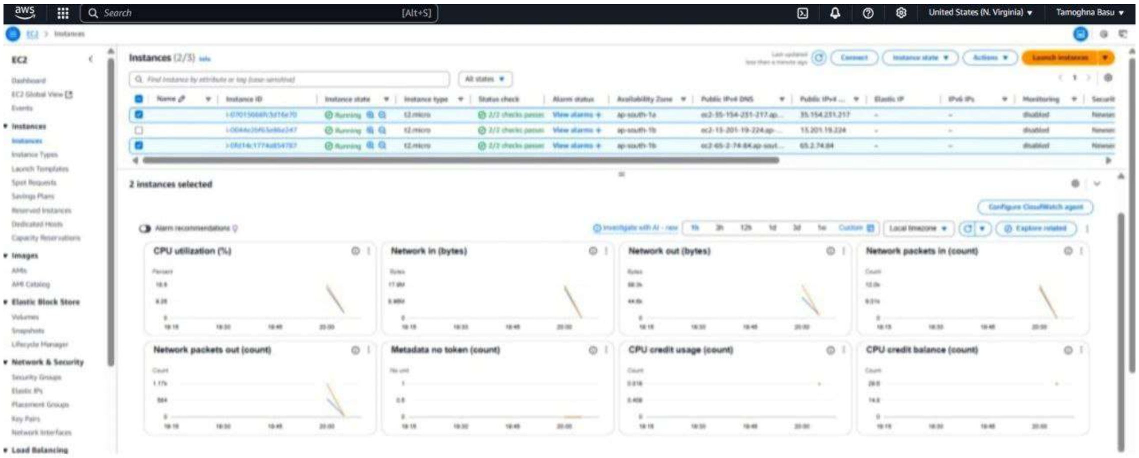


```
  GNU nano 7.2                                           ri.sh *
#!/bin/bash
while(true)
do
        echo "Inside loop"
done
```

```
                                              [ Cancelled ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo
```



Step-8:CPU utilization is checked on the monitoring section after the bash file is executed and the third instance is created as a result of it.



Instances (2/3)

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Securit
☑		i-07015684fc3d16e70	Running	t2.micro	2/2 checks passe	View alarms +	ap-south-1a	ec2-35-154-231-217.ap...	35.154.231.217	–	–	disabled	Newer
□		i-0044e2bf63e86e247	Running	t2.micro	2/2 checks passe	View alarms +	ap-south-1b	ec2-15-201-19-224.ap-...	15.201.19.224	–	–	disabled	Newer
☑		i-0fd14c1774a854787	Running	t2.micro	2/2 checks passe	View alarms +	ap-south-1b	ec2-65-2-74-84.ap-sout...	65.2.74.84	–	–	disabled	Newer

2 instances selected