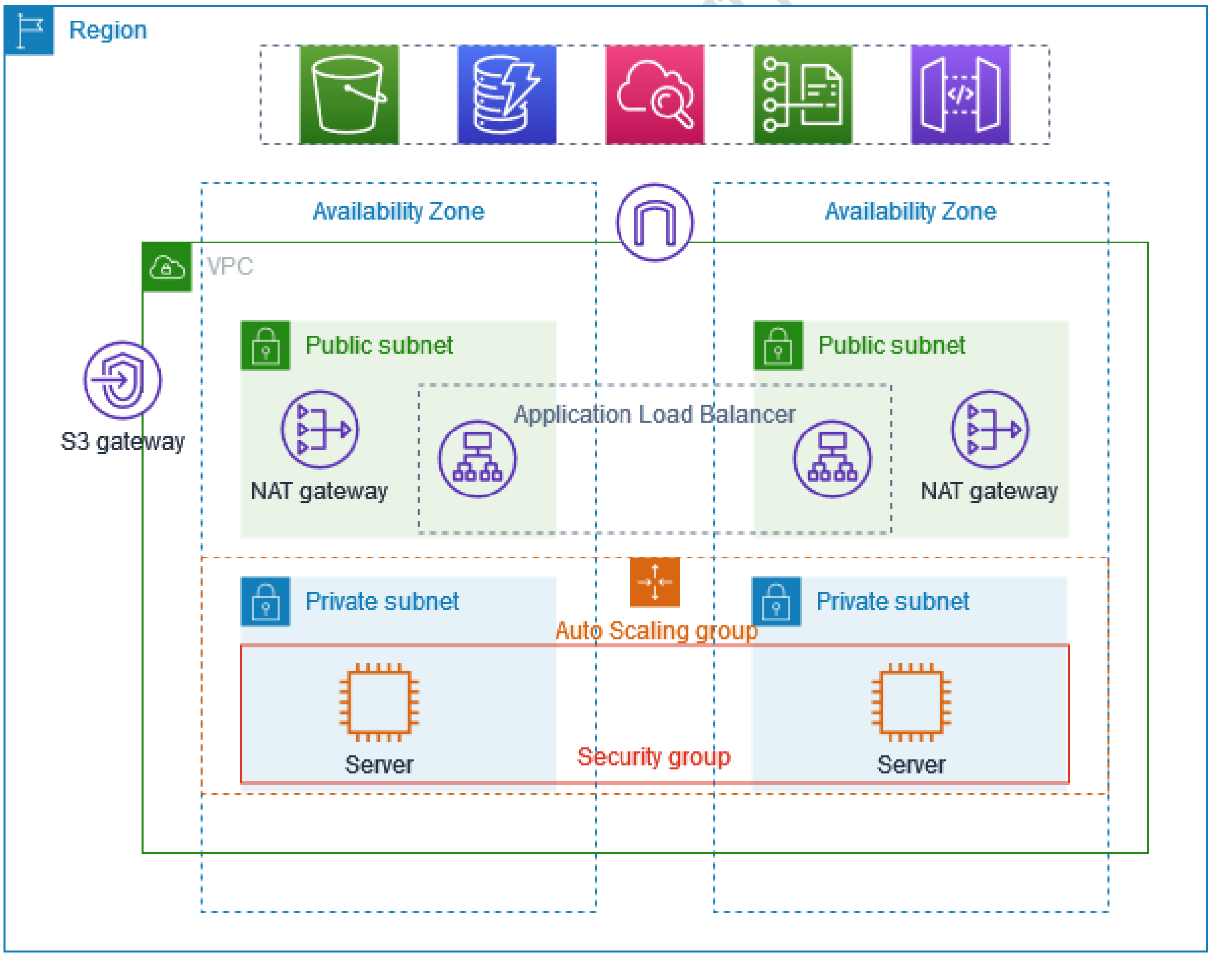
AWS 2-Tier Architecture Setup for Production Environments

VPC with public-private subnet in Production

About the Project:

This example demonstrates how to create a VPC that you can use for servers in a production environment.To improve resiliency, you deploy the servers in two Availability Zones, by using an Auto Scaling group and an Application Load Balancer. For additional security, you deploy the servers in private subnets. The servers receive requests through the load balancer. The servers can connect to the internet by using a NAT gateway. To improve resiliency, you deploy the NAT gateway in both Availability Zones.

tfolio use only



Overview:

- The VPC has public subnets and private subnets in two Availability Zones.
- Each public subnet contains a NAT gateway and a load balancer node.
- The servers run in the private subnets, are launched and terminated by using an
- Auto Scaling group, and receive traffic from the load balancer.
- The servers can connect to the internet by using the NAT gateway.

Resources required:

- Auto Scaling Group

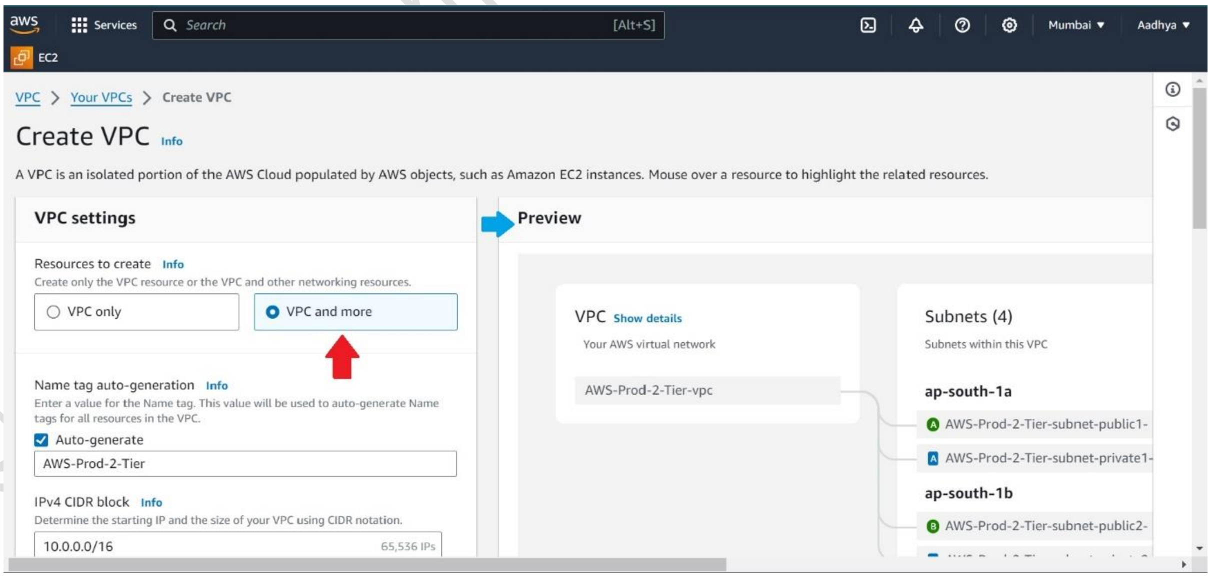
- Load Balancer

- Target Group

- Bastion Host or Jump Server

<u>Steps:</u>

1. Login into AWS console and go to VPC dashboard to create VPC.
2. Click on Create VPC.
3. In VPC settings, select VPC and more. Once you select VPC and more you can see a preview diagrammatic representation at the right side.
4. Select the name of the project – AWS-Prod-2-Tier. In IPv4 CIDR Block range as 10.0.0.0/16.



VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
○ VPC only   ● VPC and more

Name tag auto-generation Info
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.
☑ Auto-generate
AWS-Prod-2-Tier

IPv4 CIDR block Info
Determine the starting IP and the size of your VPC using CIDR notation.
10.0.0.0/16   65,536 IPs

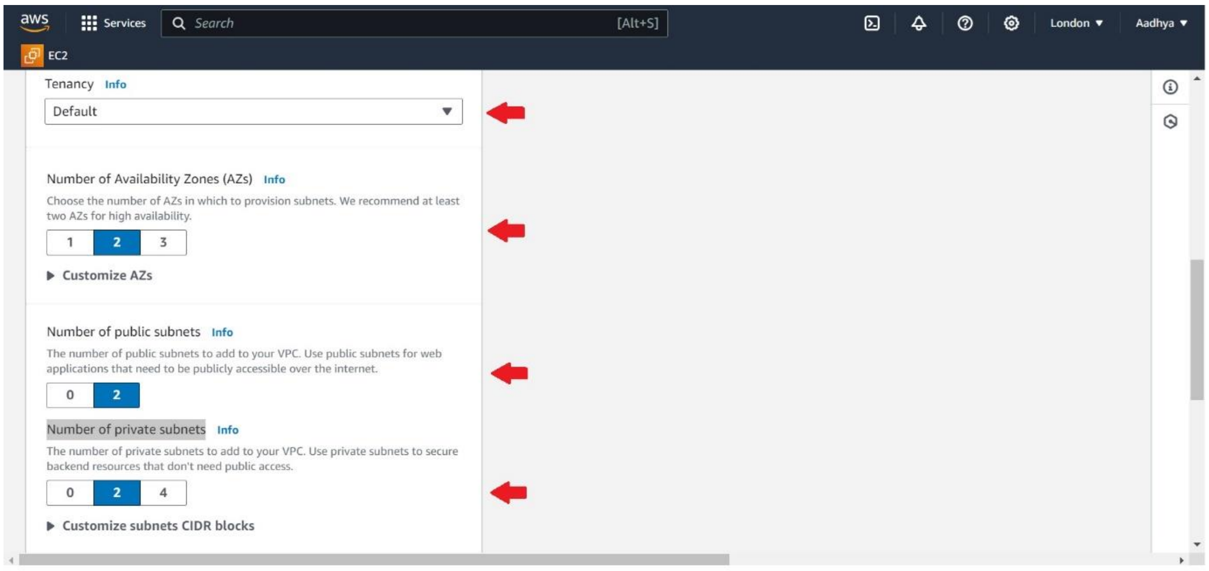
Preview

VPC Show details
Your AWS virtual network
AWS-Prod-2-Tier-vpc

Subnets (4)
Subnets within this VPC
ap-south-1a
AWS-Prod-2-Tier-subnet-public1-
AWS-Prod-2-Tier-subnet-private1-
ap-south-1b
AWS-Prod-2-Tier-subnet-public2-

5. Keep the Tenancy as Default itself.
6. Select 2 from all Number of Availability Zones (AZs), Number of public subnets & Number of private subnets.

© mo/portfolio use only


Tenancy Info

Default

Number of Availability Zones (AZs) Info

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 [✓ 2] 3

Customize AZs

Number of public subnets Info

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 [✓ 2]

Number of private subnets Info

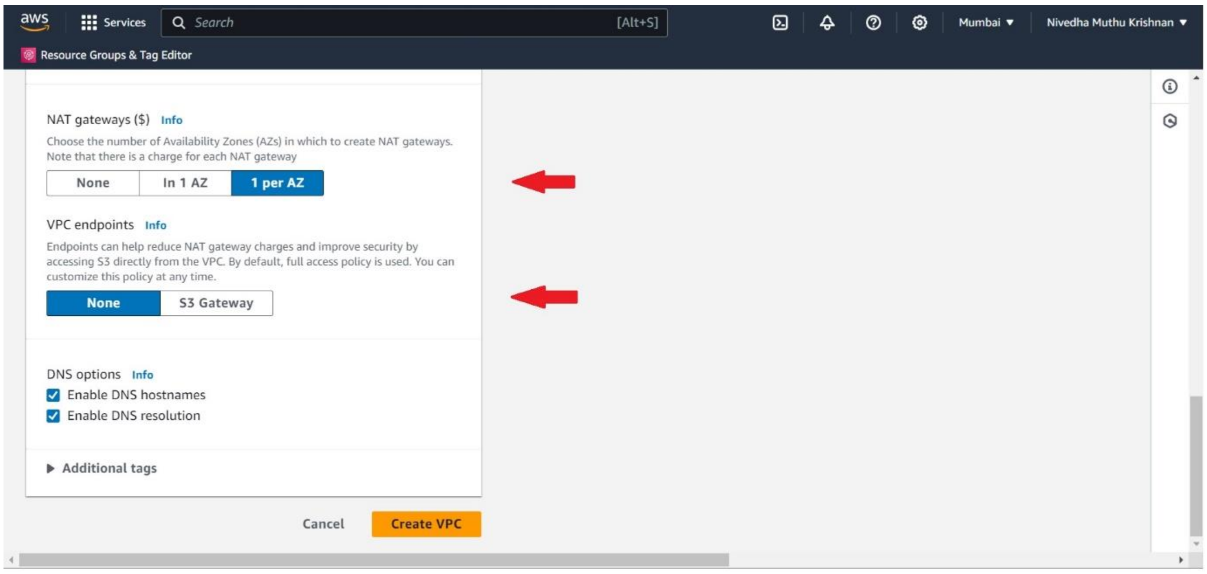
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 [✓ 2] 4

Customize subnets CIDR blocks

7. In NAT gateways, select - In 1 AZ.
8. Select None in VPC endpoints.
9. Keep all other options as same and click on Create VPC.
10. Now you can see VPC is getting created, and in while creating NAT gateway it takes time so wait till it gets completed.

© Nive | For demo/portfolio use



NAT gateways ($) Info

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

None | In 1 AZ | [✓ 1 per AZ]

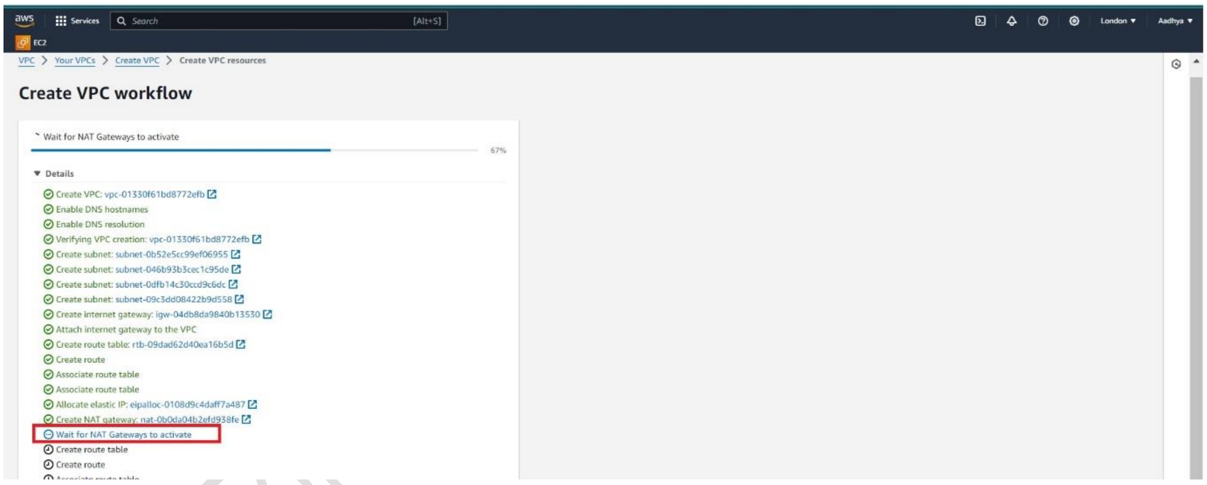
VPC endpoints Info

Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

[✓ None] | S3 Gateway

DNS options Info

- [x] Enable DNS hostnames
- [x] Enable DNS resolution



VPC > Your VPCs > Create VPC > Create VPC resources

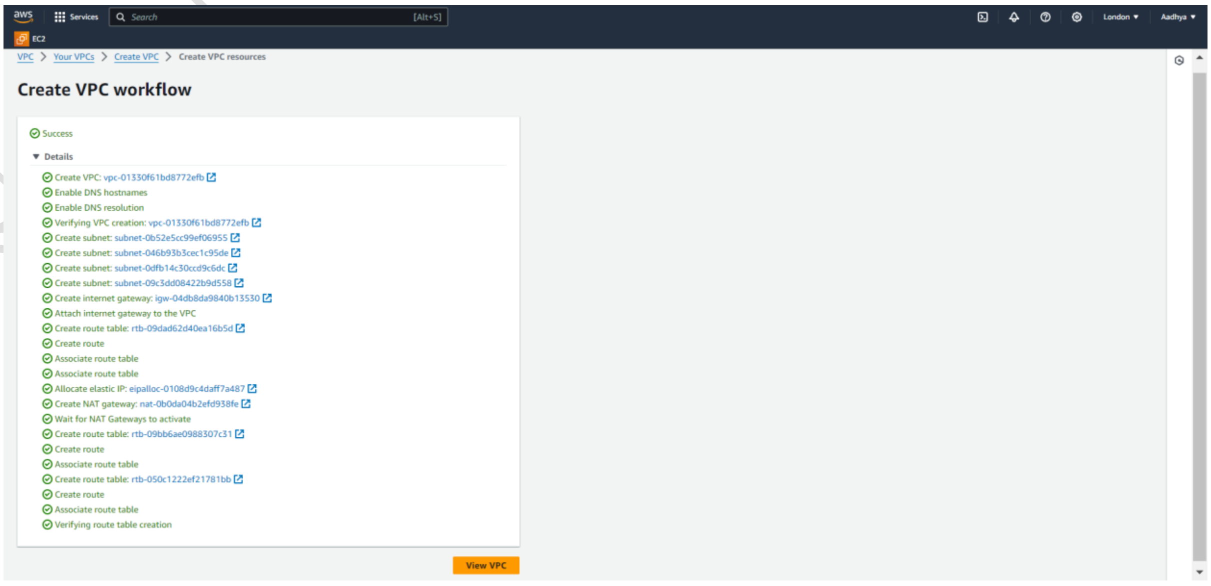
Create VPC workflow

Wait for NAT Gateways to activate 67%

Details

- Create VPC: vpc-01330f61bd8772efb
- Enable DNS hostnames
- Enable DNS resolution
- Verifying VPC creation: vpc-01330f61bd8772efb
- Create subnet: subnet-0b52e5cc99ef06955
- Create subnet: subnet-046b93b3cec1c95de
- Create subnet: subnet-0dfb14c30ccd9c6dc
- Create subnet: subnet-09c3dd08422b9d558
- Create internet gateway: igw-04db8da9840b13530
- Attach internet gateway to the VPC
- Create route table: rtb-09dad62d40ea16b5d
- Create route
- Associate route table
- Associate route table
- Allocate elastic IP: eipalloc-0108d9c4daff7a487
- Create NAT gateway: nat-0b0da04b2efd938fe
- Wait for NAT Gateways to activate
- Create route table
- Create route

11. Now you can see VPC has been created successfully. Just check if all we created is present or not.



VPC > Your VPCs > Create VPC > Create VPC resources

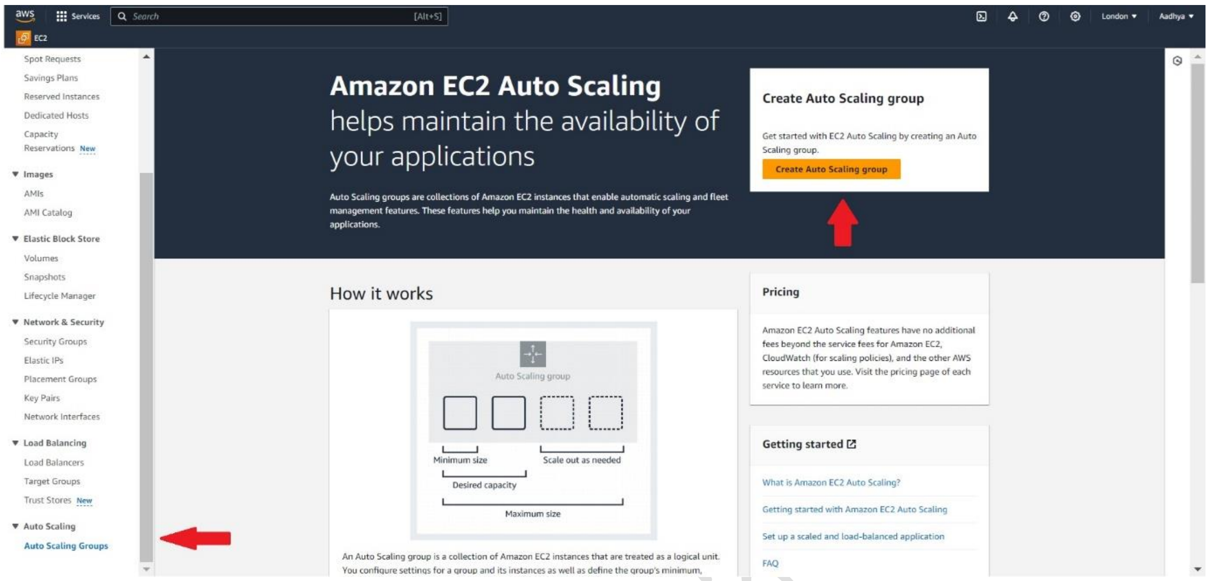
Create VPC workflow

Success

Details

- Create VPC: vpc-01330f61bd8772efb
- Enable DNS hostnames
- Enable DNS resolution
- Verifying VPC creation: vpc-01330f61bd8772efb
- Create subnet: subnet-0b52e5cc99ef06955
- Create subnet: subnet-046b93b3cec1c95de
- Create subnet: subnet-0dfb14c30ccd9c6dc
- Create subnet: subnet-09c3dd08422b9d558
- Create internet gateway: igw-04db8da9840b13530
- Attach internet gateway to the VPC
- Create route table: rtb-09dad62d40ea16b5d
- Create route
- Associate route table
- Associate route table
- Allocate elastic IP: eipalloc-0108d9c4daff7a487
- Create NAT gateway: nat-0b0da04b2efd938fe
- Wait for NAT Gateways to activate
- Create route table: rtb-09bb6ae0988307c31
- Create route
- Associate route table
- Create route table: rtb-050c1222ef21781bb
- Create route
- Associate route table
- Verifying route table creation

12. Now go to EC2 dashboard, to find ASG(Auto Scaling Group) and then click on 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.

Create Auto Scaling group

How it works

An Auto Scaling group is a collection of Amazon EC2 instances that are treated as a logical unit. You configure settings for a group and its instances as well as define the group's minimum,

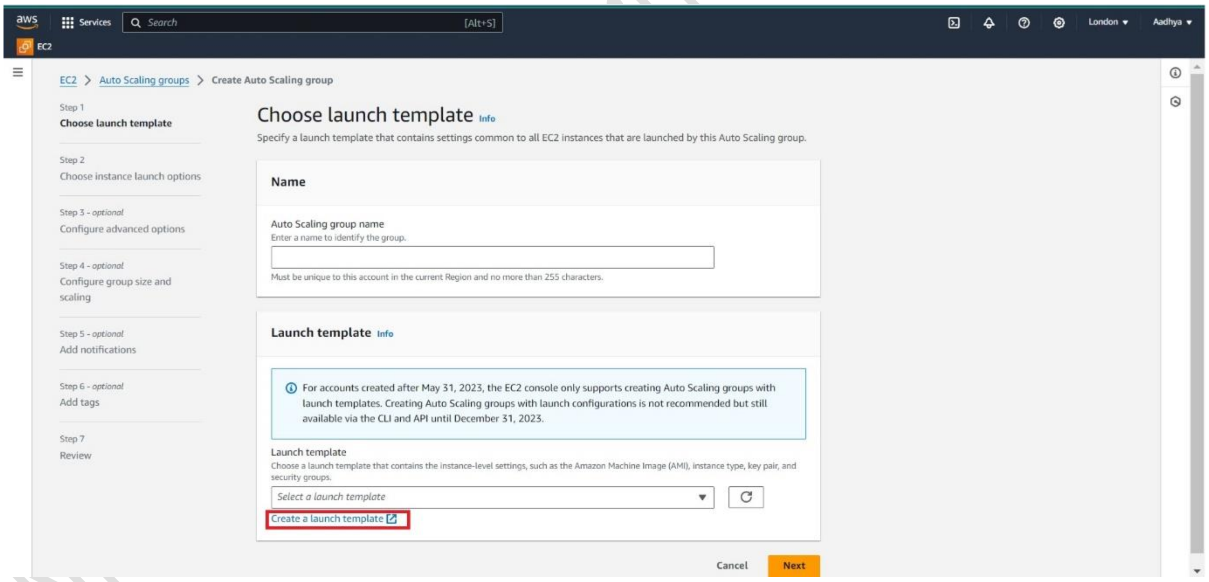
Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Getting started

What is Amazon EC2 Auto Scaling?
Getting started with Amazon EC2 Auto Scaling
Set up a scaled and load-balanced application
FAQ

13. Once you are in to ASG page, we have to create ASG template so click on Create a launch template it will automatically redirects to a new tab to create a new template.



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.

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.

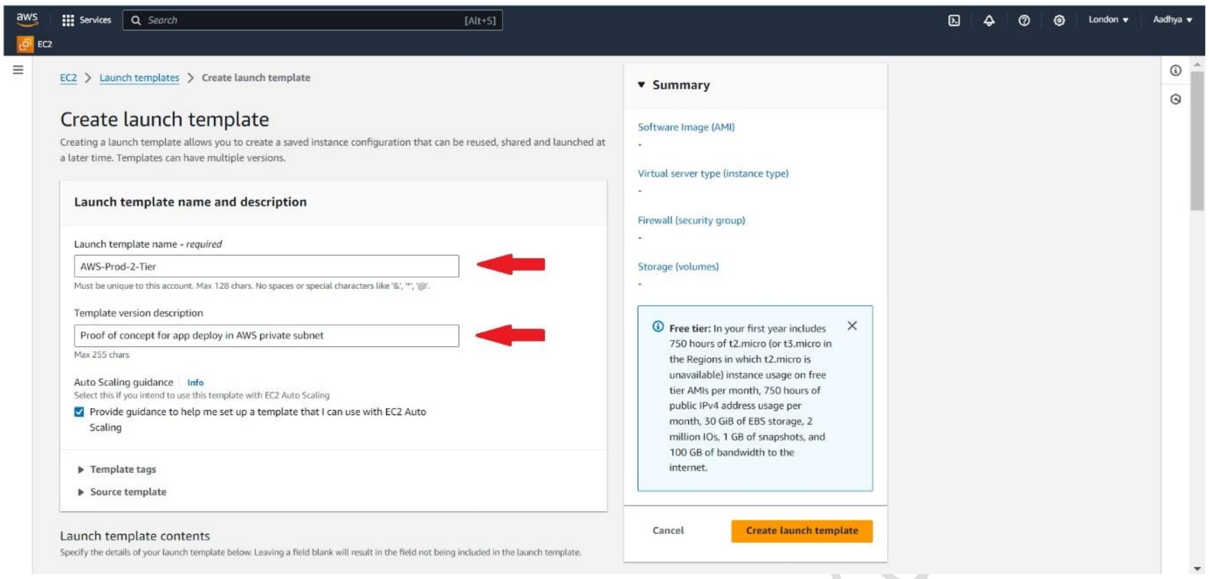
Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Select a launch template

Create a launch template

Cancel Next

14. In ASG Template page, Launch template name required- AWS-Prod-2-Tier, Template version description- Proof of concept for app deploy in AWS private subnet.



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*

AWS-Prod-2-Tier

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

Template version description

Proof of concept for app deploy in AWS private subnet

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.

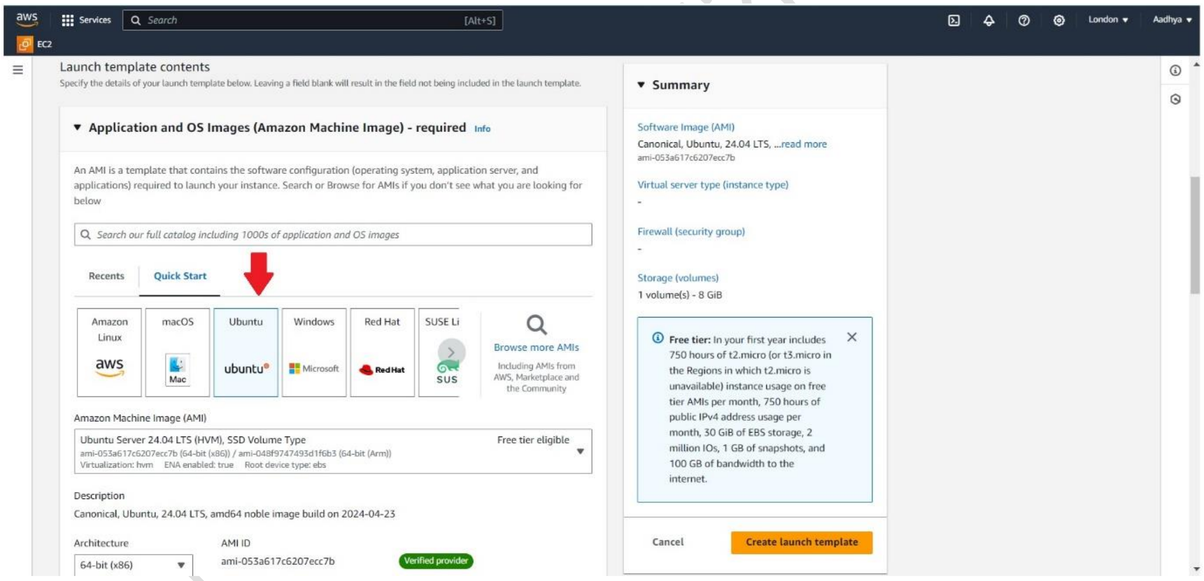
▾ **Summary**

Software Image (AMI)
-
Virtual server type (instance type)
-
Firewall (security group)
-
Storage (volumes)
-

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.

Cancel | Create launch template

15. Select Application and OS Images (Amazon Machine Image) – required- as Ubuntu AMI.



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

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Li | 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-053a617c6207ecc7b (64-bit (x86)) / ami-048f9747493d1f6b3 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Canonical, Ubuntu, 24.04 LTS, amd64 noble image build on 2024-04-23

Architecture: 64-bit (x86)

AMI ID: ami-053a617c6207ecc7b — Verified provider

▾ **Summary**

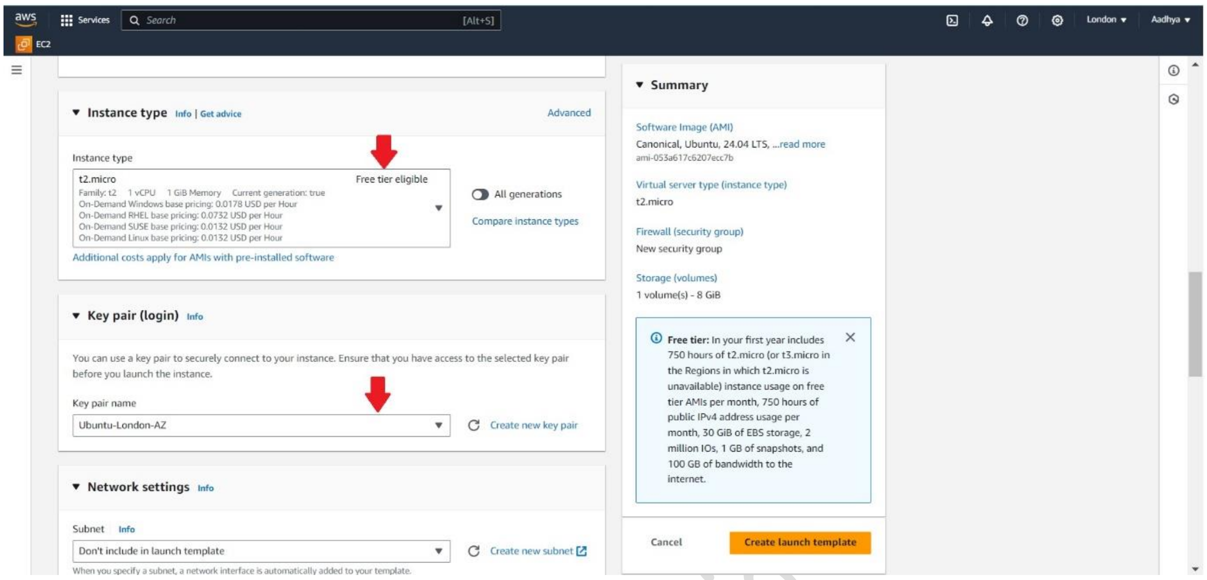
Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-053a617c6207ecc7b
Virtual server type (instance type)
-
Firewall (security group)
-
Storage (volumes)
1 volume(s) - 8 GiB

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.

Cancel | Create launch template

© Nive

16. In Instance type select t2.micro. Select the available keypair or you can also create new.



Instance type

Instance type: t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0178 USD per Hour
On-Demand RHEL base pricing: 0.0732 USD per Hour
On-Demand SUSE base pricing: 0.0132 USD per Hour
On-Demand Linux base pricing: 0.0132 USD per Hour
Additional costs apply for AMIs with pre-installed software

Key pair (login)

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: Ubuntu-London-AZ

Network settings

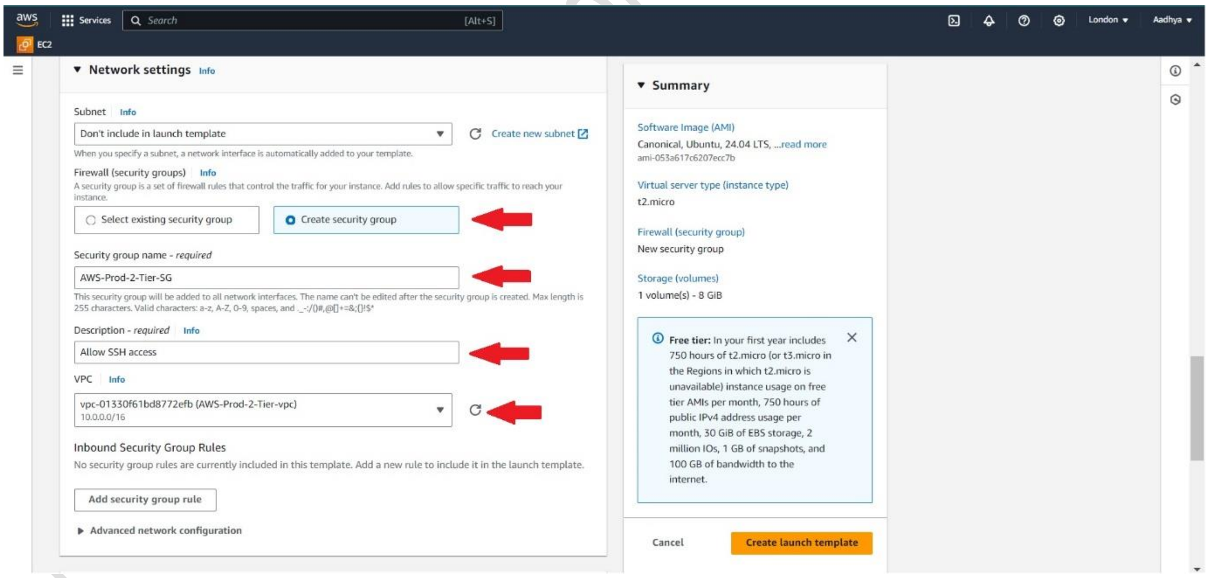
Subnet: Don't include in launch template
When you specify a subnet, a network interface is automatically added to your template.

Summary

Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ...read more
ami-053a617c6207ecc7b
Virtual server type (instance type): t2.micro
Firewall (security group): New 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.

17. In Network settings, don’t touch any subnet configuration. Select Create security group and give the sg name as AWS-Prod-2-Tier-SG, description as Allow SSH access. Select the VPC(AWS-Prod-2-Tier) which we created earlier.



Network settings

Subnet: Don't include in launch template
When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups)
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 group name - *required*: AWS-Prod-2-Tier-SG
This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

Description - *required*: Allow SSH access

VPC: vpc-01330f61bd8772efb (AWS-Prod-2-Tier-vpc)
10.0.0.0/16

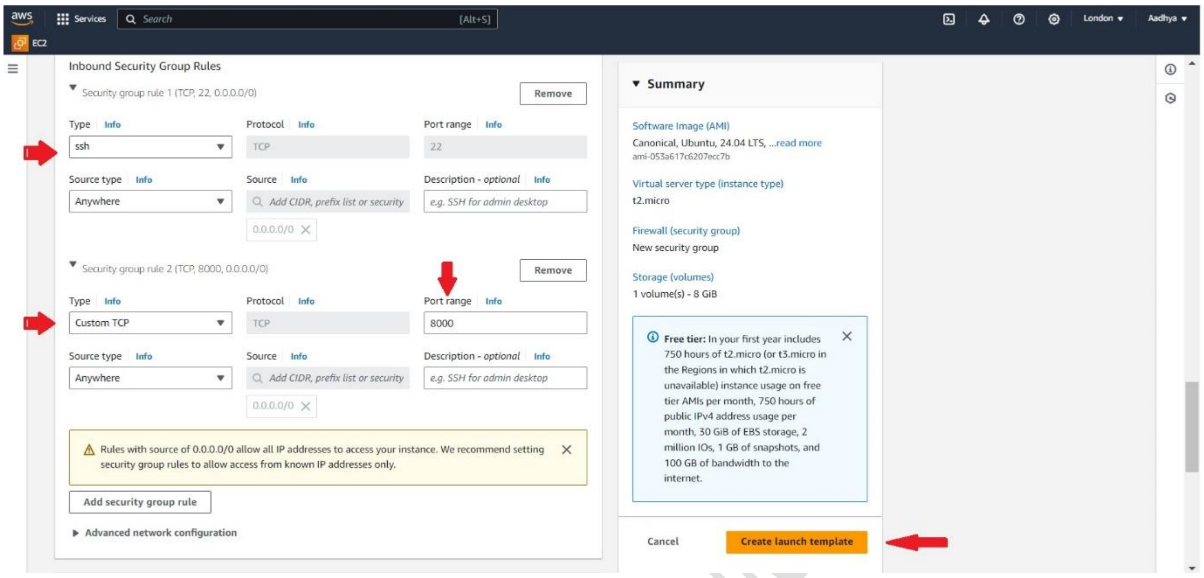
Inbound Security Group Rules
No security group rules are currently included in this template. Add a new rule to include it in the launch template.

Summary

Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ...read more
ami-053a617c6207ecc7b
Virtual server type (instance type): t2.micro
Firewall (security group): New 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.

18. In Inbound Security Group Rules, add SSH(from anywhere) and custom TCP with port range 8000(from anywhere). Leave all other configurations untouched and click on Create launch template.



aws Services Search [Alt+S] London ▾ Aadhya ▾
EC2

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

Type Info: ssh
Protocol Info: TCP
Port range Info: 22
Source type Info: Anywhere
Source Info: Add CIDR, prefix list or security
0.0.0.0/0 ✕
Description - optional Info: e.g. SSH for admin desktop

▼ Security group rule 2 (TCP, 8000, 0.0.0.0/0) Remove

Type Info: Custom TCP
Protocol Info: TCP
Port range Info: 8000
Source type Info: Anywhere
Source Info: Add CIDR, prefix list or security
0.0.0.0/0 ✕
Description - optional Info: e.g. SSH for admin desktop

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. ✕

Add security group rule

▶ Advanced network configuration

▼ Summary

Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-053a617c6207ecc7b

Virtual server type (instance type)
t2.micro

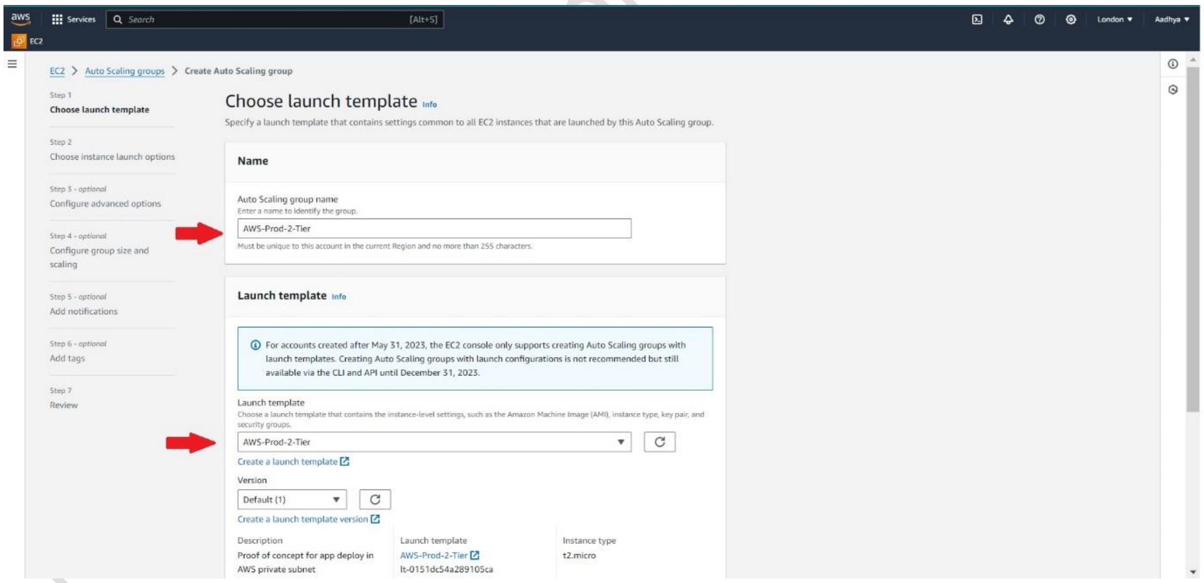
Firewall (security group)
New security group

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. ✕

Cancel | Create launch template

19. Now go back to ASG page and refresh once so that you can see the created template in the options. Give the Auto Scaling group name - AWS-Prod-2-Tier and then click on Next.



aws Services Search [Alt+S] London ▾ Aadhya ▾
EC2

EC2 > Auto Scaling groups > Create Auto Scaling group

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

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.
AWS-Prod-2-Tier
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

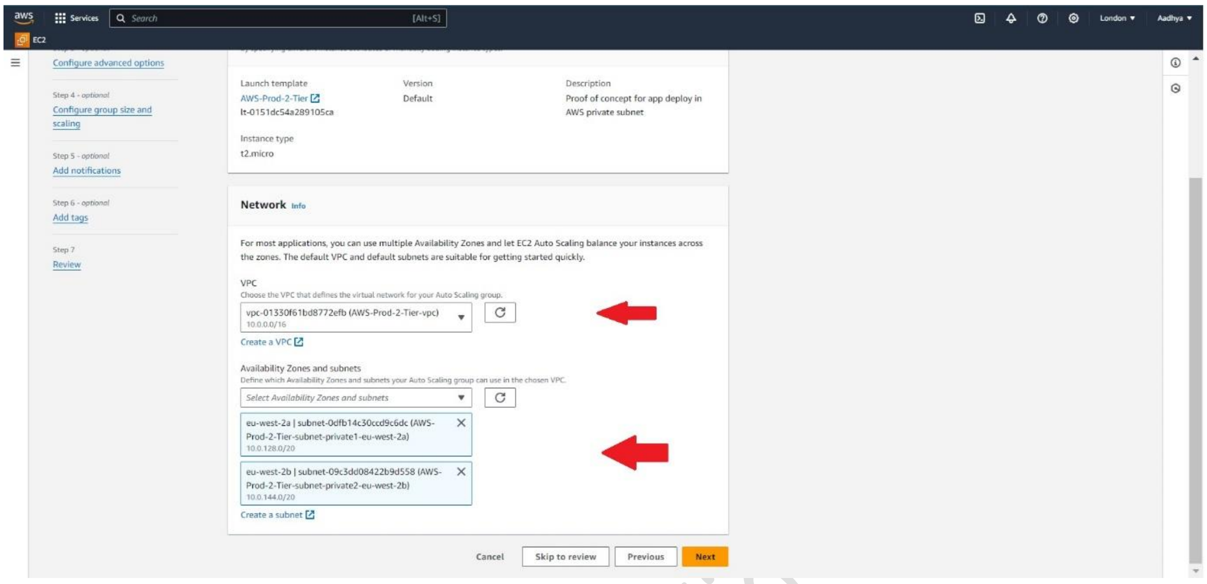
ⓘ For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

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.
AWS-Prod-2-Tier
Create a launch template

Version
Default (1)
Create a launch template version

Description	Launch template	Instance type
Proof of concept for app deploy in AWS private subnet	AWS-Prod-2-Tier lt-0151dc54a289105ca	t2.micro

20. In Network, select the VPC created and in Availability Zones and subnets – select the 2 available private subnets. Now click on Next.



Configure advanced options

Step 4 - *optional*
Configure group size and scaling

Step 5 - *optional*
Add notifications

Step 6 - *optional*
Add tags

Step 7
Review

Launch template	Version	Description
AWS-Prod-2-Tier lt-0151dc54a289105ca	Default	Proof of concept for app deploy in AWS private subnet

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-01330f61bd8772efb (AWS-Prod-2-Tier-vpc)
10.0.0.0/16
Create a VPC

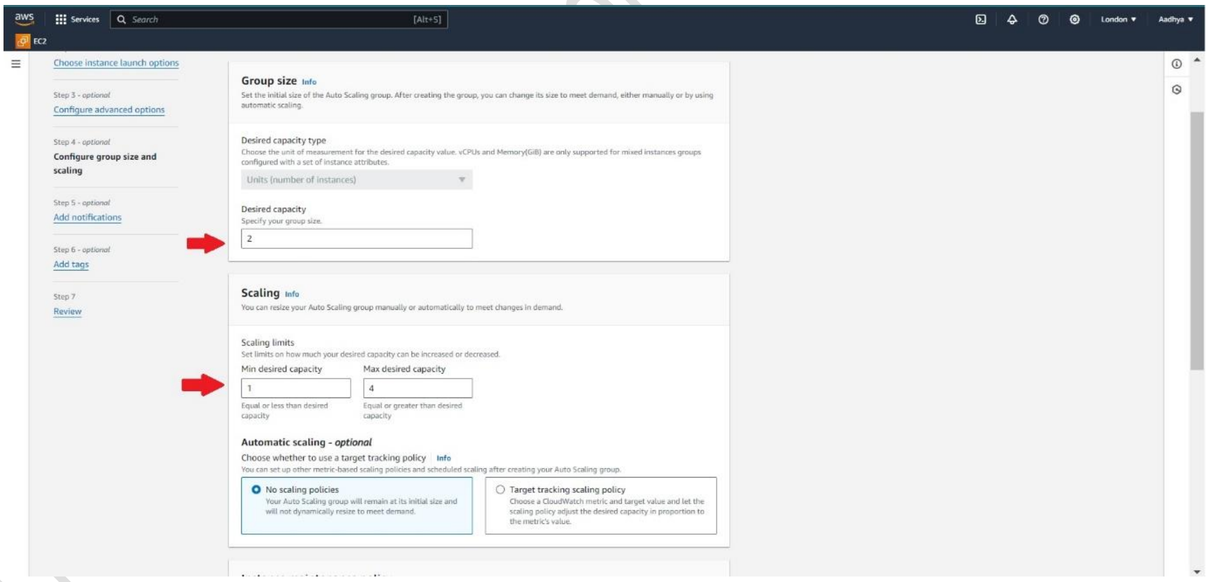
Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets

eu-west-2a | subnet-0dfb14c30ccd9c6dc (AWS-Prod-2-Tier-subnet-private1-eu-west-2a)
10.0.128.0/20

eu-west-2b | subnet-09c3dd08422b9d558 (AWS-Prod-2-Tier-subnet-private2-eu-west-2b)
10.0.144.0/20

Create a subnet

21. Configure advanced options – optional page don't change any. Leave as it is and click on Next.
22. Give 2 in Desired capacity and in Scaling Min as 1 and maximum as 4. Don't touch automatic scaling configurations. Click on Next.



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

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.

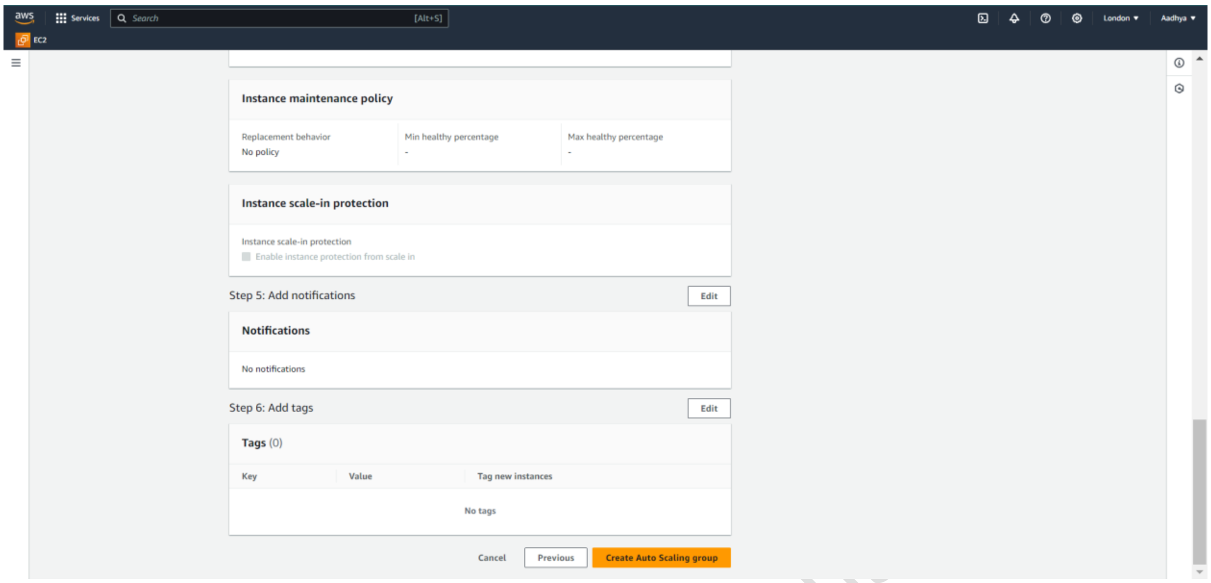
Min desired capacity	Max desired capacity
1	4
Equal or less than desired capacity	Equal or greater than desired capacity

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.

23. Don’t touch any other options and directly click on Create Auto Scaling group.



Instance maintenance policy

Replacement behavior	Min healthy percentage	Max healthy percentage
No policy	-	-

Instance scale-in protection

Instance scale-in protection
Enable instance protection from scale in

Step 5: Add notifications

Notifications

No notifications

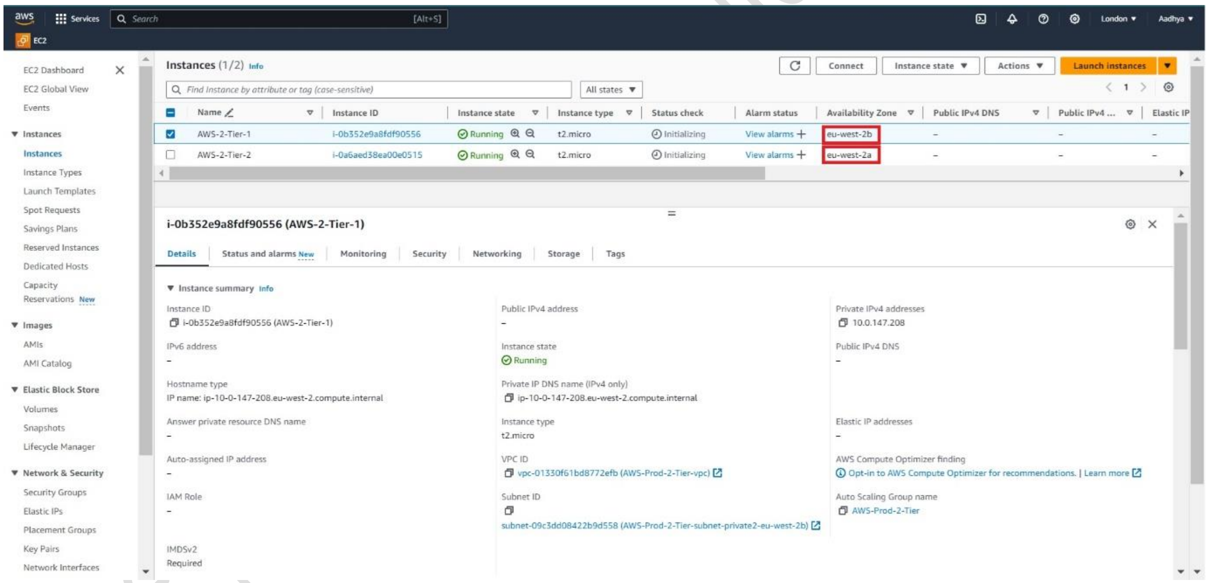
Step 6: Add tags

Tags (0)

Key	Value	Tag new instances

No tags

24. Make sure if the instances are created and also see if it is created in two different AZ’s.



Instances (1/2)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
AWS-2-Tier-1	i-0b352e9a8fdf90556	Running	t2.micro	Initializing	View alarms	eu-west-2b	–	–	–
AWS-2-Tier-2	i-0a6aed38ea00e0515	Running	t2.micro	Initializing	View alarms	eu-west-2a	–	–	–

i-0b352e9a8fdf90556 (AWS-2-Tier-1)

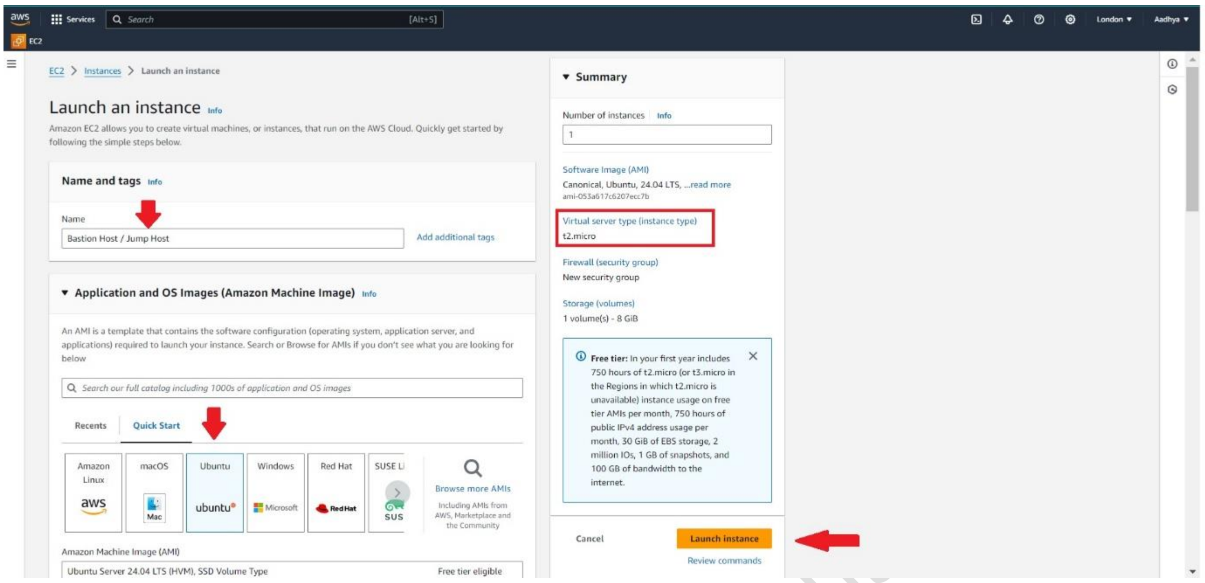
Instance summary

Instance ID: i-0b352e9a8fdf90556 (AWS-2-Tier-1)	Public IPv4 address: –	Private IPv4 addresses: 10.0.147.208
IPv6 address: –	Instance state: Running	Public IPv4 DNS: –
Hostname type: IP name: ip-10-0-147-208.eu-west-2.compute.internal	Private IP DNS name (IPv4 only): ip-10-0-147-208.eu-west-2.compute.internal	
Answer private resource DNS name: –	Instance type: t2.micro	Elastic IP addresses: –
Auto-assigned IP address: –	VPC ID: vpc-01330f61bd8772efb (AWS-Prod-2-Tier-vpc)	AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role: –	Subnet ID: subnet-09c3dd08422b9d558 (AWS-Prod-2-Tier-subnet-private2-eu-west-2b)	Auto Scaling Group name: AWS-Prod-2-Tier
IMDSv2: Required		

Now to login into the private instances we need a bastion host.

25. Go to EC2 Dashboard, click on Launch Instance. Give the name as Bastion Host / Jump Host, select ubuntu AMI, instance type t2.micro and provide

keypair.



EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: Bastion Host / Jump Host

Application and OS Images (Amazon Machine Image) 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

Amazon Machine Image (AMI): Ubuntu Server 24.04 LTS (HVM), SSD Volume Type — Free tier eligible

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ...read more; ami-053a617c6207ecc7b

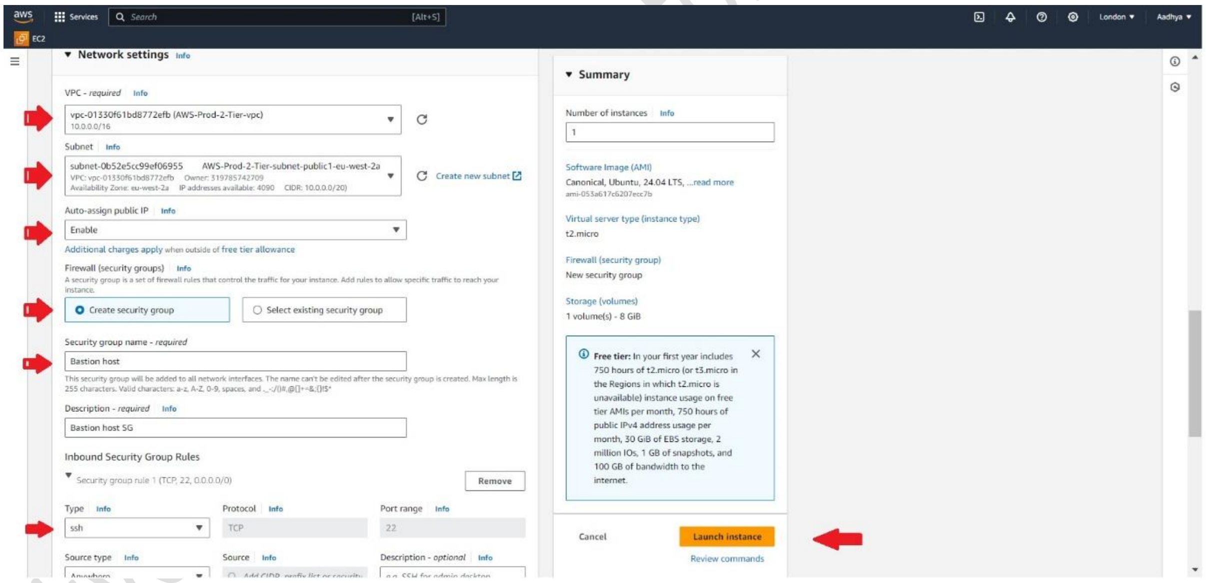
Virtual server type (instance type): t2.micro

Firewall (security group): New 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.

26. In network settings, select the created VPC, make sure it is in public subnet, Auto-assign public IP – Enable, Create security group and make sure ssh is added. Now click on Launch instance.



Network settings Info

VPC - required: vpc-01330f61bd8772efb (AWS-Prod-2-Tier-vpc) 10.0.0.0/16

Subnet: subnet-0b52e5cc99ef06955 AWS-Prod-2-Tier-subnet-public1-eu-west-2a; VPC: vpc-01330f61bd8772efb Owner: 319785742709; Availability Zone: eu-west-2a IP addresses available: 4090 CIDR: 10.0.0.0/20)

Auto-assign public IP: Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups): Create security group

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.

Security group name - required: Bastion host

Description - required: Bastion host SG

Inbound Security Group Rules

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type: ssh Protocol: TCP Port range: 22

Source type: Anywhere

Summary

Number of instances: 1

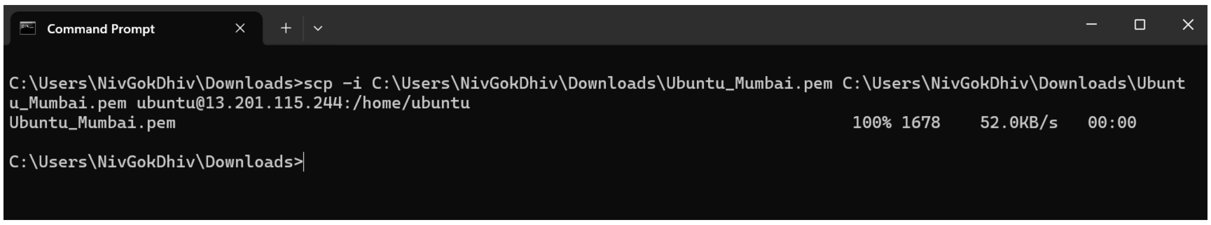
Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ...read more; ami-053a617c6207ecc7b

Virtual server type (instance type): t2.micro

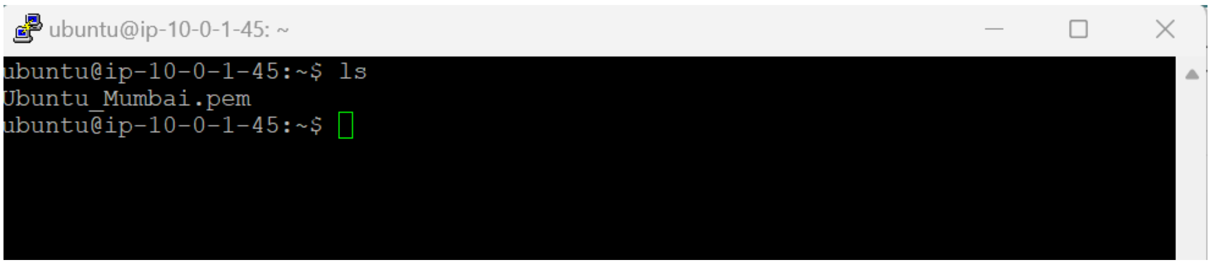
Firewall (security group): New security group

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

27. Now open CMD and type the below command to
scp -i C:\Users\NivGokDhiv\Downloads\Ubuntu_Mumbai.pem C:\Users\NivGokDhiv\Downloads\Ubuntu_Mumbai.pem ubuntu@13.201.115.244:/home/ubuntu



```
C:\Users\NivGokDhiv\Downloads>scp -i C:\Users\NivGokDhiv\Downloads\Ubuntu_Mumbai.pem C:\Users\NivGokDhiv\Downloads\Ubunt
u_Mumbai.pem ubuntu@13.201.115.244:/home/ubuntu
Ubuntu_Mumbai.pem                                                                   100% 1678     52.0KB/s   00:00

C:\Users\NivGokDhiv\Downloads>
```



```
ubuntu@ip-10-0-1-45:~$ ls
Ubuntu_Mumbai.pem
ubuntu@ip-10-0-1-45:~$
```

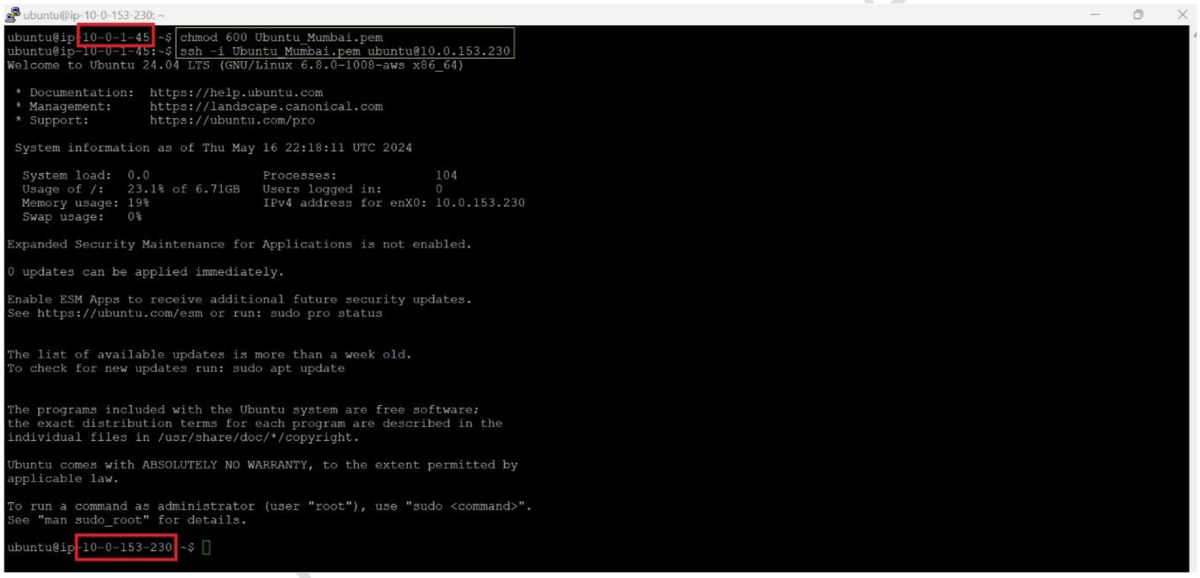
28. Change the permission of the pem file into 600 using the below command.

    Chmod 600 Ubuntu_Mumbai.pem

29. Now to login into the private instance, copy any one of the private instance IP- 10.0.153.230 and do ssh from bastion host with the below command via putty.

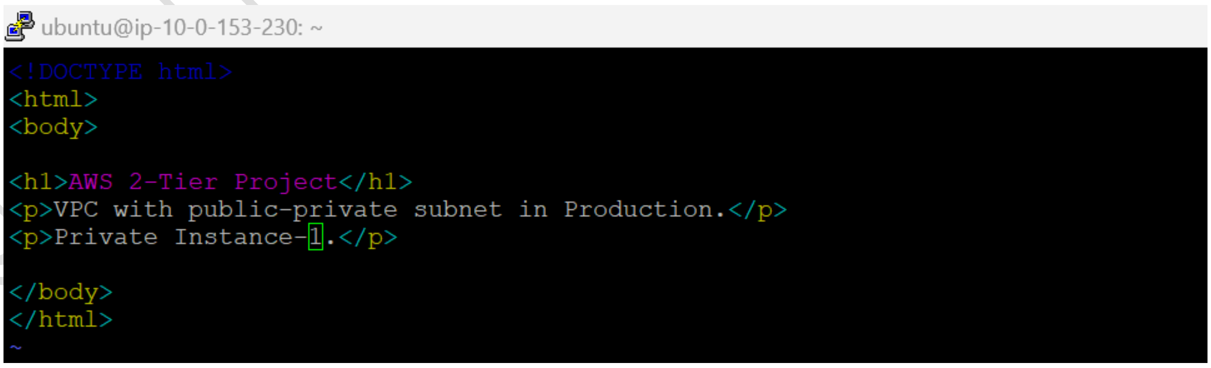
    ssh -I Ubuntu_Mumbai.pem ubuntu@10.0.153.230

30. Now you can notice that the IP has been changed from Bastion to private IP as shown in the below screenshot.



```
ubuntu@ip-10-0-1-45:~$ chmod 600 Ubuntu_Mumbai.pem
ubuntu@ip-10-0-1-45:~$ ssh -i Ubuntu_Mumbai.pem ubuntu@10.0.153.230
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1008-aws x86_64)

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

 System information as of Thu May 16 22:18:11 UTC 2024

  System load:  0.0               Processes:             104
  Usage of /:   23.1% of 6.71GB   Users logged in:       0
  Memory usage: 19%               IPv4 address for enX0: 10.0.153.230
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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


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


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

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

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

ubuntu@ip-10-0-153-230:~$
```

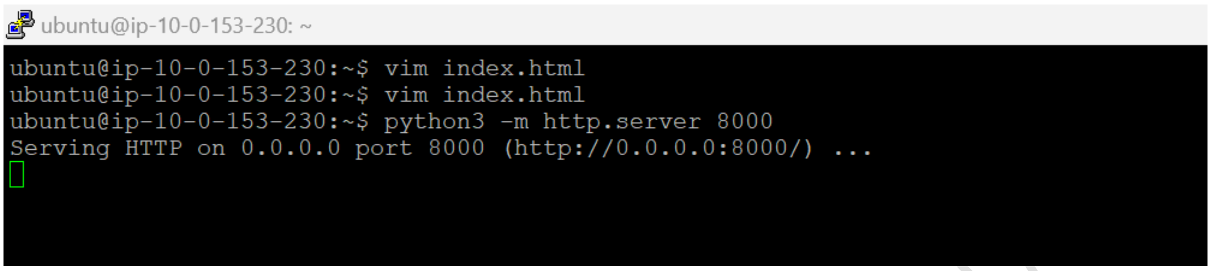
31. Now once u logged inside the private IP, create a index.html file using the below command and give the contents of HTML inside it.



```
<!DOCTYPE html>
<html>
<body>

<h1>AWS 2-Tier Project</h1>
<p>VPC with public-private subnet in Production.</p>
<p>Private Instance-1.</p>

</body>
</html>
~
```

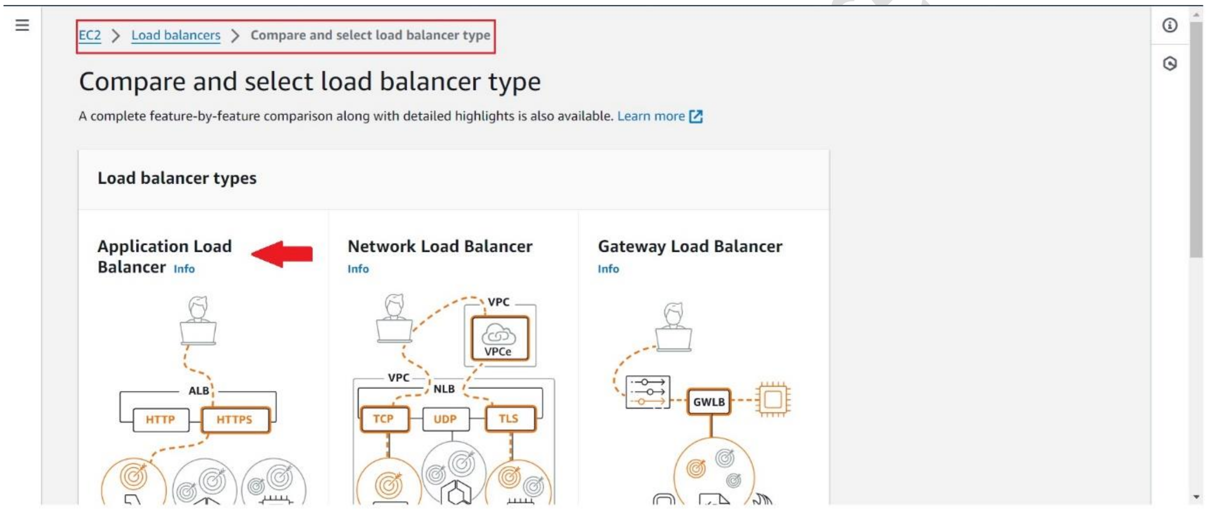
32. Now install the python application and run port 8000 in it using the below command.

python3 -m http.server 8000



```
ubuntu@ip-10-0-153-230: ~
ubuntu@ip-10-0-153-230:~$ vim index.html
ubuntu@ip-10-0-153-230:~$ vim index.html
ubuntu@ip-10-0-153-230:~$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

33. Now to balance the load between two private instances now we have to create Load Balancers. Go to EC2 Dashboard, select Load balancers from the left panel, select Application Load Balancer and click on Next.



EC2 > Load balancers > Compare and select load balancer type

Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. Learn more

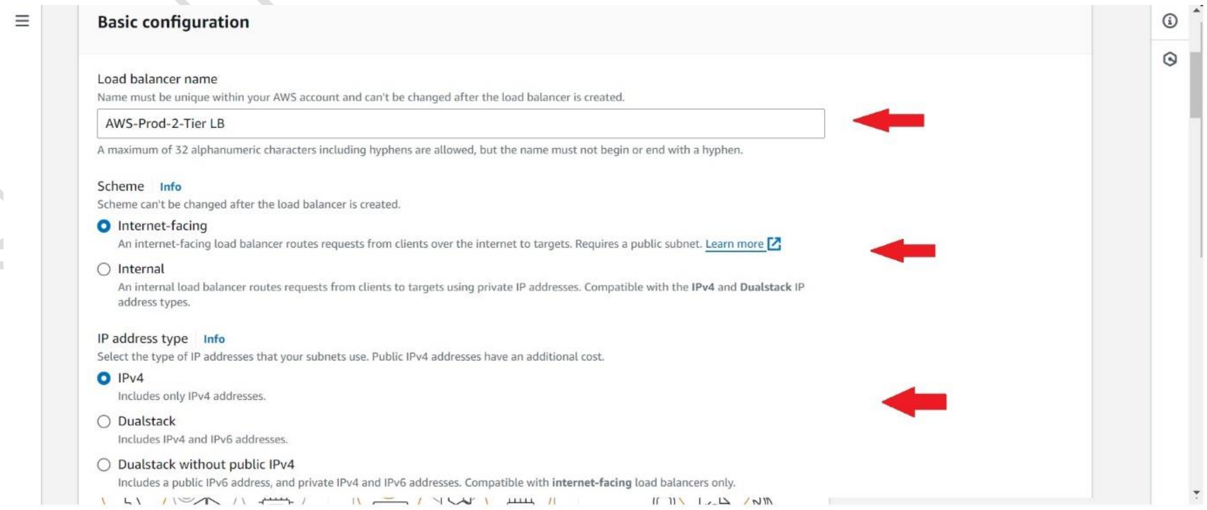
Load balancer types

Application Load Balancer Info

Network Load Balancer Info

Gateway Load Balancer Info

34. Give the Load balancer name AWS-Prod-2Tier-LB, keep the scheme as internet-facing as LB should be always public. IP address type is IPv4.



Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.

AWS-Prod-2-Tier LB

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme Info
Scheme can't be changed after the load balancer is created.

[✓ Internet-facing]
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. Learn more

Internal
An internal load balancer routes requests from clients to targets using private IP addresses. Compatible with the **IPv4** and **Dualstack** IP address types.

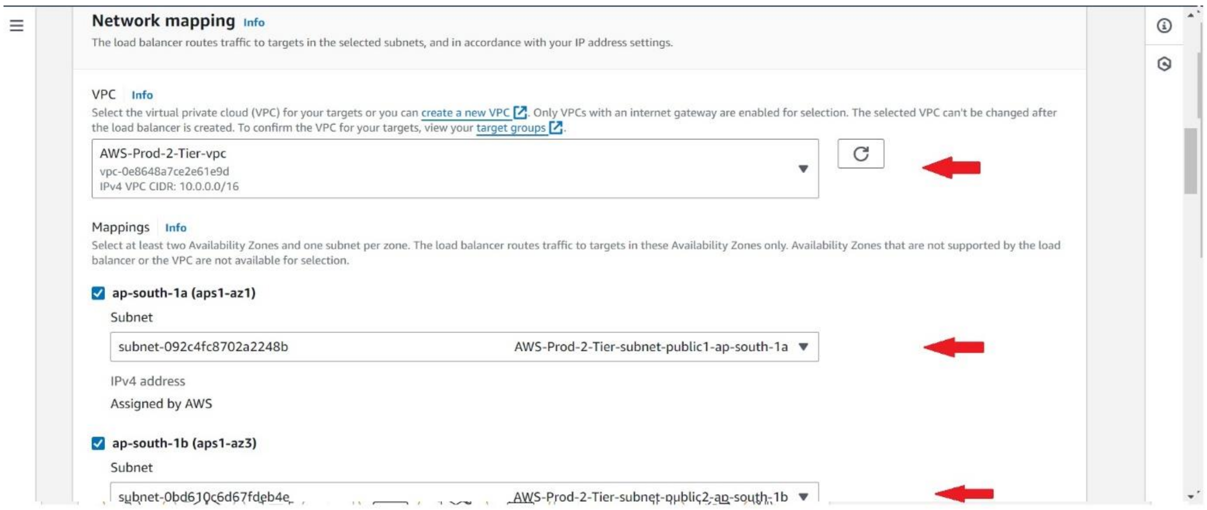
IP address type Info
Select the type of IP addresses that your subnets use. Public IPv4 addresses have an additional cost.

[✓ IPv4]
Includes only IPv4 addresses.

Dualstack
Includes IPv4 and IPv6 addresses.

Dualstack without public IPv4
Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with **internet-facing** load balancers only.

35. In Network Mapping, select the created VPC. Select the available mappings and make sure both mappins are in public subnets , if not change it public subnet from the dropdown.


Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info

Select the virtual private cloud (VPC) for your targets or you can create a new VPC. Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.

AWS-Prod-2-Tier-vpc
vpc-0e8648a7ce2e61e9d
IPv4 VPC CIDR: 10.0.0.0/16

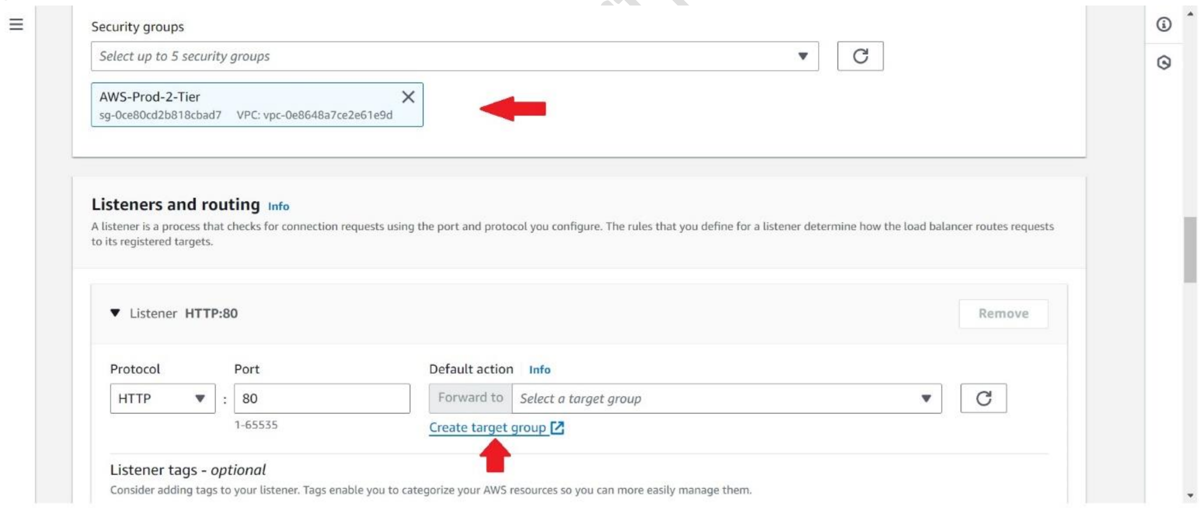
Mappings Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

ap-south-1a (aps1-az1)
Subnet
subnet-092c4fc8702a2248b AWS-Prod-2-Tier-subnet-public1-ap-south-1a
IPv4 address
Assigned by AWS

ap-south-1b (aps1-az3)
Subnet
subnet-0bd610c6d67fdeb4e AWS-Prod-2-Tier-subnet-public2-ap-south-1b

36. Select the VPC security group. In Listners and Routing, Click on Create target group.


Security groups
Select up to 5 security groups
AWS-Prod-2-Tier
sg-0ce80cd2b818cbad7 VPC: vpc-0e8648a7ce2e61e9d

Listeners and routing Info

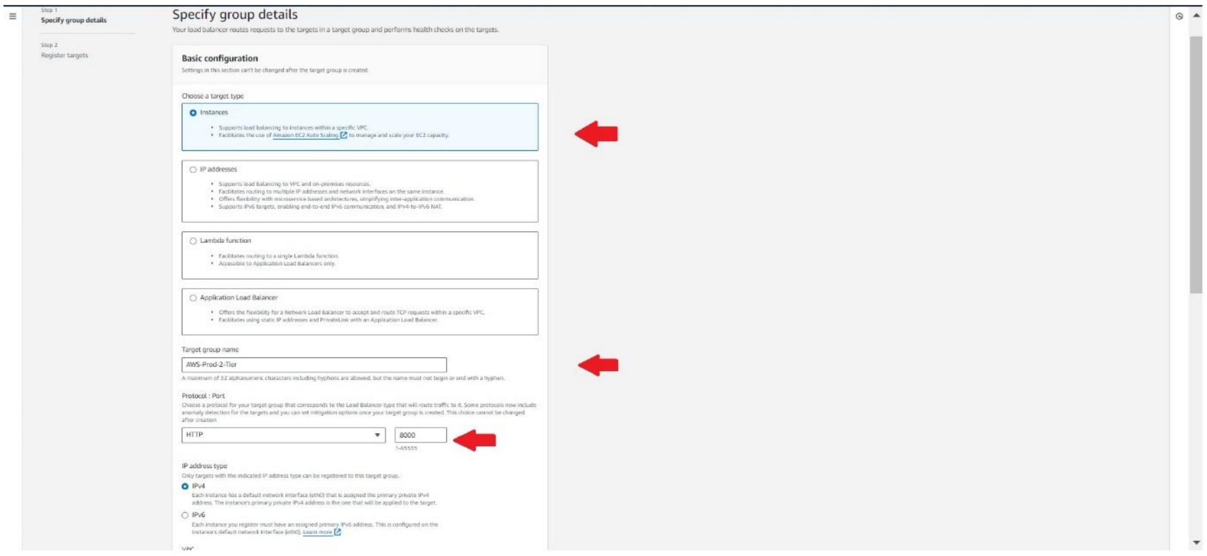
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80 Remove

Protocol	Port	Default action Info
HTTP	80	Forward to Select a target group
	1-65535	Create target group

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

37. In the target group page, specify the basic configuration as Instances, in Protocol-Port select HTTP and give 8000 in it.


Step 1
Specify group details
Step 2
Register targets

Specify group details
Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Basic configuration
Settings in this section can't be changed after the target group is created.

Choose a target type

[✓ Instances]
- Supports load balancing to instances within a specific VPC.
- Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.

IP addresses
- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function
- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

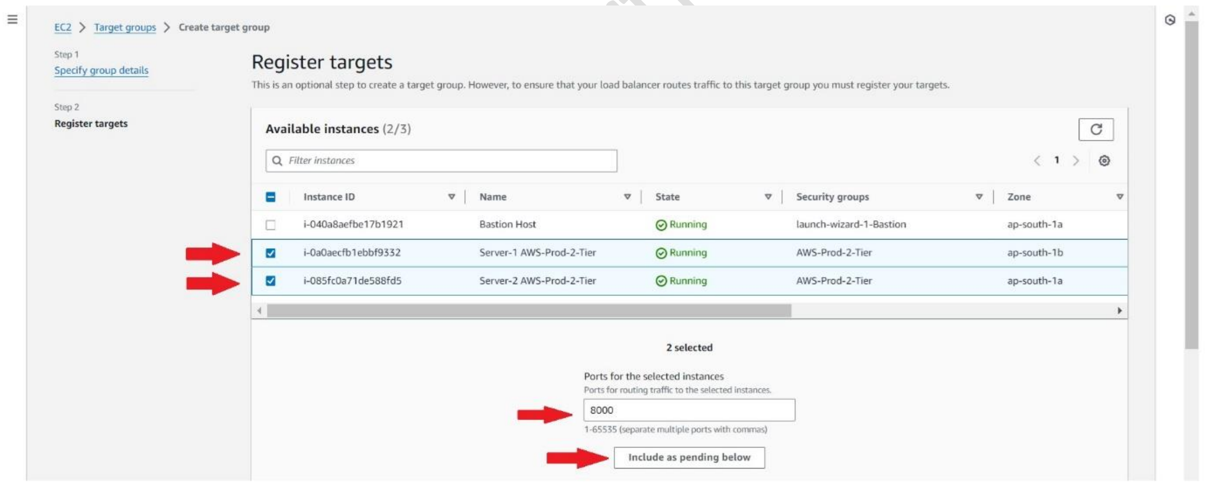
Application Load Balancer
- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name
AWS-Prod-2-Tier
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol : Port
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation
HTTP
8000
1-65535

IP address type
Only targets with the indicated IP address type can be registered to this target group.
[✓ IPv4]
Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.
IPv6
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more

38. Select the two instances other than bastion host and click on include as pending below. In the ports for the selected instances give port number as 8000.


EC2 > Target groups > Create target group

Step 1
Specify group details
Step 2
Register targets

Register targets
This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

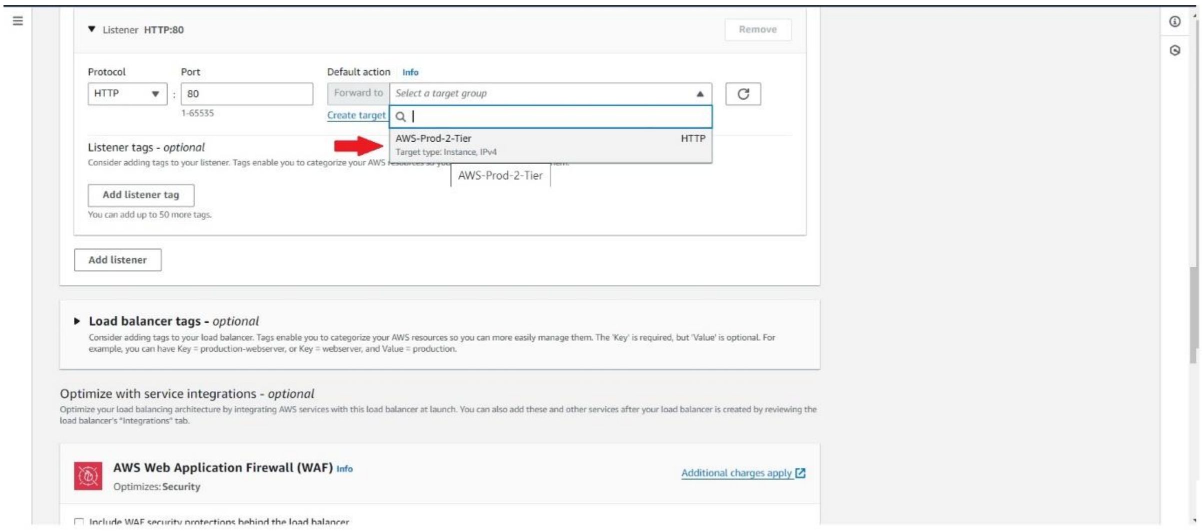
Available instances (2/3)

	Instance ID	Name	State	Security groups	Zone
	i-040a8aefbe17b1921	Bastion Host	Running	launch-wizard-1-Bastion	ap-south-1a
✓	i-0a0aecfb1ebbf9332	Server-1 AWS-Prod-2-Tier	Running	AWS-Prod-2-Tier	ap-south-1b
✓	i-085fc0a71de588fd5	Server-2 AWS-Prod-2-Tier	Running	AWS-Prod-2-Tier	ap-south-1a

2 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.
8000
1-65535 (separate multiple ports with commas)

39. Now to LB page again refresh it and select the created target group, and create the Load balancer directly without changing anyother options.



▼ Listener HTTP:80 | Remove

Protocol: HTTP | Port: 80 (1-65535) | Default action Info: Forward to | Select a target group

Create target

AWS-Prod-2-Tier HTTP
Target type: Instance, IPv4

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

Add listener tag
You can add up to 50 more tags.

Add listener

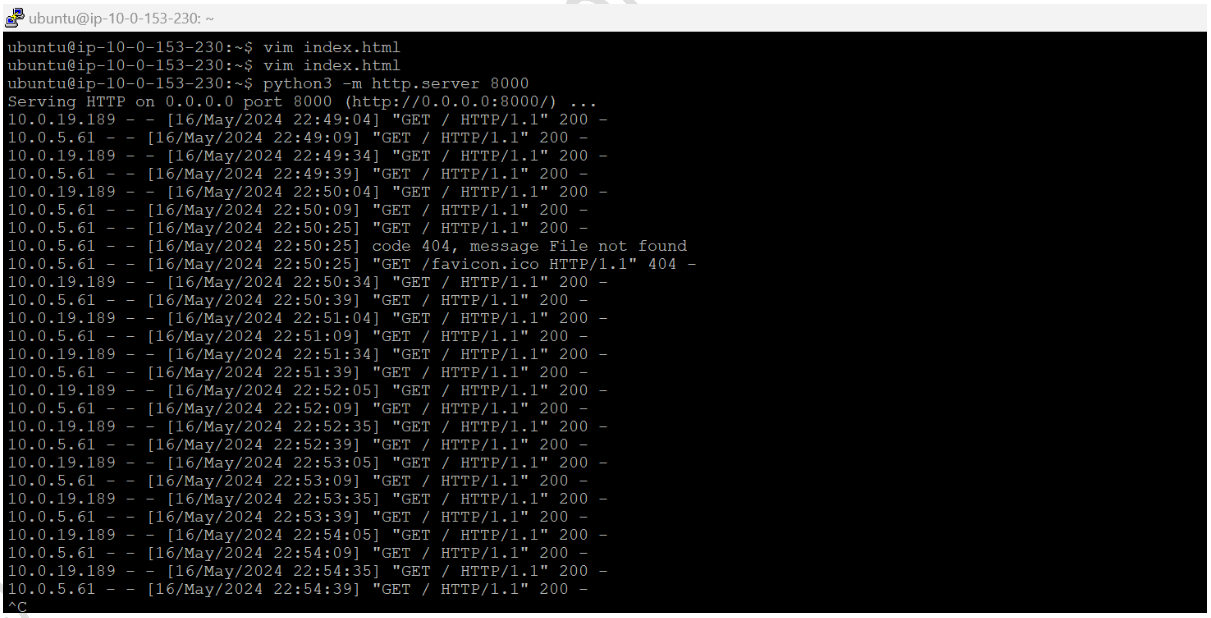
▸ **Load balancer tags** - *optional*
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.

Optimize with service integrations - *optional*
Optimize your load balancing architecture by integrating AWS services with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the load balancer's "Integrations" tab.

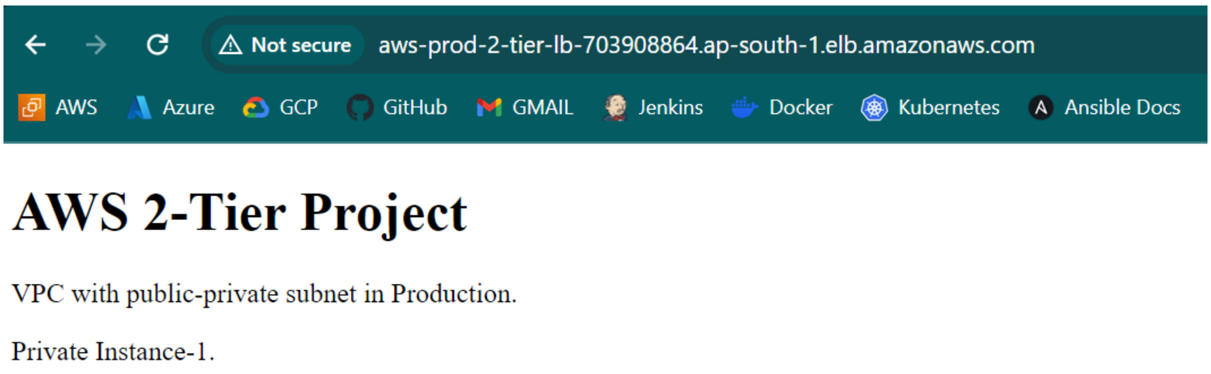
AWS Web Application Firewall (WAF) Info
Optimizes: Security
Additional charges apply

☐ Include WAF security protections behind the load balancer

40. Now once the LB is active copy the DNS name and paste it in web page now you can see the index.html files. Now once you see the output repeat the same in server 2 also.



```
ubuntu@ip-10-0-153-230: ~
ubuntu@ip-10-0-153-230:~$ vim index.html
ubuntu@ip-10-0-153-230:~$ vim index.html
ubuntu@ip-10-0-153-230:~$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.0.19.189 - - [16/May/2024 22:49:04] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:49:09] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:49:34] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:49:39] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:50:04] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:50:09] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:50:25] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:50:25] code 404, message File not found
10.0.5.61 - - [16/May/2024 22:50:25] "GET /favicon.ico HTTP/1.1" 404 -
10.0.19.189 - - [16/May/2024 22:50:34] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:50:39] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:51:04] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:51:09] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:51:34] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:51:39] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:52:05] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:52:09] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:52:35] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:52:39] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:53:05] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:53:09] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:53:35] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:53:39] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:54:05] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:54:09] "GET / HTTP/1.1" 200 -
10.0.19.189 - - [16/May/2024 22:54:35] "GET / HTTP/1.1" 200 -
10.0.5.61 - - [16/May/2024 22:54:39] "GET / HTTP/1.1" 200 -
^C
```

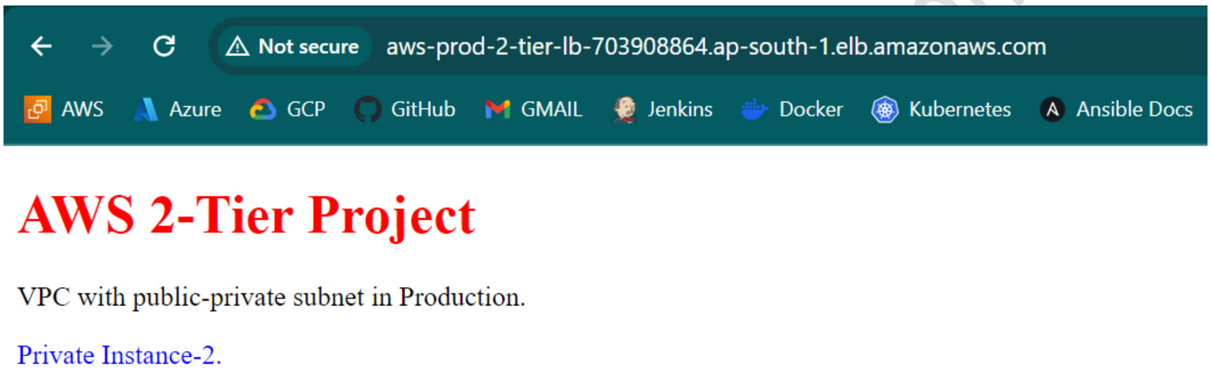

Not secure aws-prod-2-tier-lb-703908864.ap-south-1.elb.amazonaws.com

AWS Azure GCP GitHub GMAIL Jenkins Docker Kubernetes Ansible Docs

AWS 2-Tier Project

VPC with public-private subnet in Production.

Private Instance-1.


Not secure aws-prod-2-tier-lb-703908864.ap-south-1.elb.amazonaws.com

AWS Azure GCP GitHub GMAIL Jenkins Docker Kubernetes Ansible Docs

AWS 2-Tier Project

VPC with public-private subnet in Production.

Private Instance-2.

Deletion of Resources:

41. Delete the LB(along with the created Target group)
42. Delete the ASG(along with the created Template)
43. Terminate all 3 instances.
44. Now coming to delete the VPC follow the below steps-
45. NAT Gateway
46. Then directly delete the VPC(It will automaticall delete the Subnets, SG and SG)
47. Don’t forget to release the Elastic IP.

© Nive | For demo/p only