DEPLOYING HIGHLY AVAILABLE, SCALABLE AND SECURE WEBSITE ON AWS

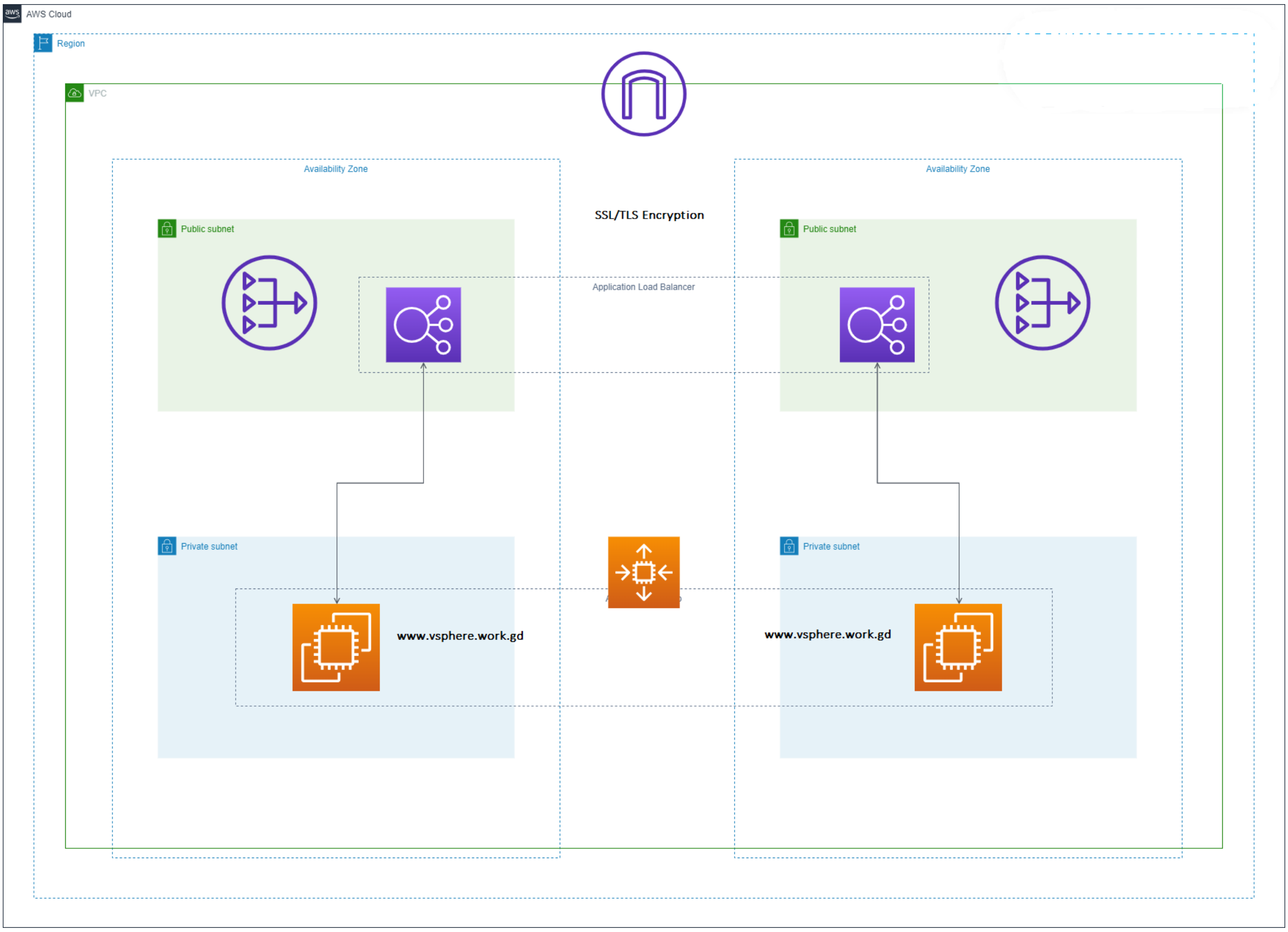
VINAY SHARMA

Table of Contents

Deploying Highly Available, Scalable and Secure Website on AWS

Diagram: High Availability and Secure Web Application on AWS

This diagram illustrates the architecture of my web application deployed on AWS, featuring load balancing, multi-AZ deployment, a custom VPC with private subnets, and enhanced security measures including a custom domain and SSL certificate.



Project Description: High Availability, Scalability and Security of Web Application on AWS

In this project, I have deployed a sample website on AWS, implementing a robust
architecture to ensure high availability, security, and scalability. The key components and configurations are as follows:

1. **Load Balancing:** To evenly distribute incoming traffic and enhance fault tolerance, I have employed an AWS Elastic Load Balancer (ELB). This ensures that user requests are efficiently handled by the available servers.
2. **Multi-AZ Deployment:** The web application is hosted across multiple Availability Zones (AZs) within a region. This setup guarantees high availability and resilience, as it mitigates the impact of an AZ failure.
3. **Custom VPC:** A custom Virtual Private Cloud (VPC) has been created to provide network isolation and enhanced security. This allows for fine-grained control over the network environment.
4. **Private Subnets:** The servers hosting the website are placed in private subnets, enhancing security by restricting direct access from the internet. These subnets are configured to route traffic through the load balancer, which is the only publicly accessible endpoint.

5. **Security Groups and NACLs:** Security groups and Network Access Control Lists (NACLs) are configured to control inbound and outbound traffic at both the instance and subnet level, ensuring that only legitimate traffic reaches the servers.

6. **Domain and SSL Certificate:** A custom domain has been purchased for the website, providing a professional and memorable address for users.
   Additionally, an SSL certificate has been generated and implemented to secure the data transmission between the users and the application, ensuring that all communications are encrypted and trustworthy.

7. **Access through Load Balancer:** The web application is only accessible through the load balancer’s public DNS, providing a single entry point and further securing the backend servers.

This architecture not only ensures high availability and fault tolerance but also secures the application by minimizing direct exposure to the internet and controlling access strictly through the load balancer. The addition of a custom domain and SSL certificate further enhances the user experience by providing a secure and reliable connection.

In this project, I got a chance to work on the following technologies:

1. AWS EC2

2. AWS Auto Scaling Group (ASG)

3. AWS Application Load Balancer (ASG)

4. AWS VPC

5. Domain Name System (DNS)

6. Private Subnet

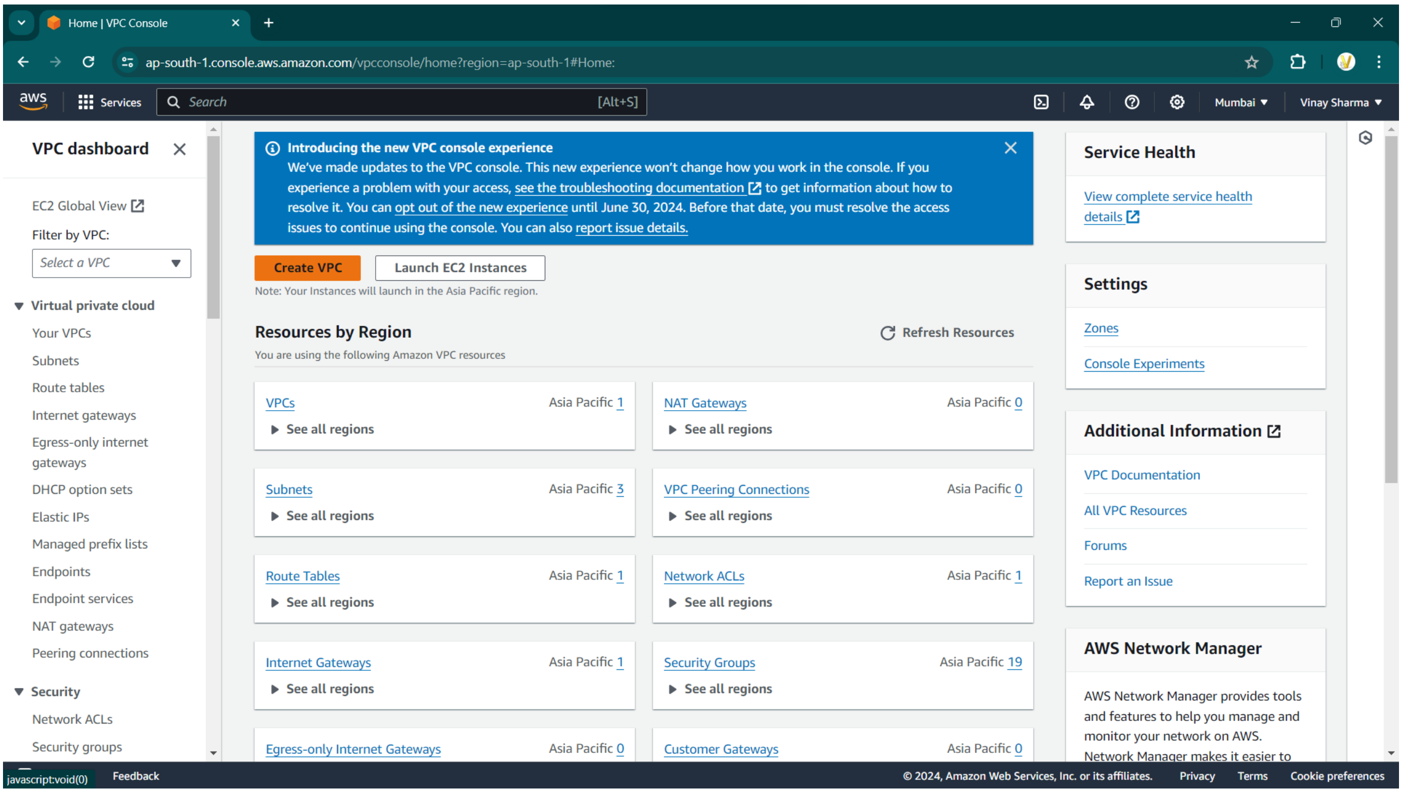
6. SSL Certificates

Practical Implementation:

Let's Create Custom VPC (Virtual Private Cloud):

VPC (Virtual Private Cloud): A VPC is a virtual network that closely resembles a traditional network that you'd operate in your own data center.

A **Custom VPC** in AWS is a user-defined Virtual Private Cloud that allows you to configure and manage your own isolated network within the AWS cloud.



Introducing the new VPC console experience
We've made updates to the VPC console. This new experience won't change how you work in the console. If you experience a problem with your access, see the troubleshooting documentation to get information about how to resolve it. You can opt out of the new experience until June 30, 2024. Before that date, you must resolve the access issues to continue using the console. You can also report issue details.

Note: Your Instances will launch in the Asia Pacific region.

Resources by Region

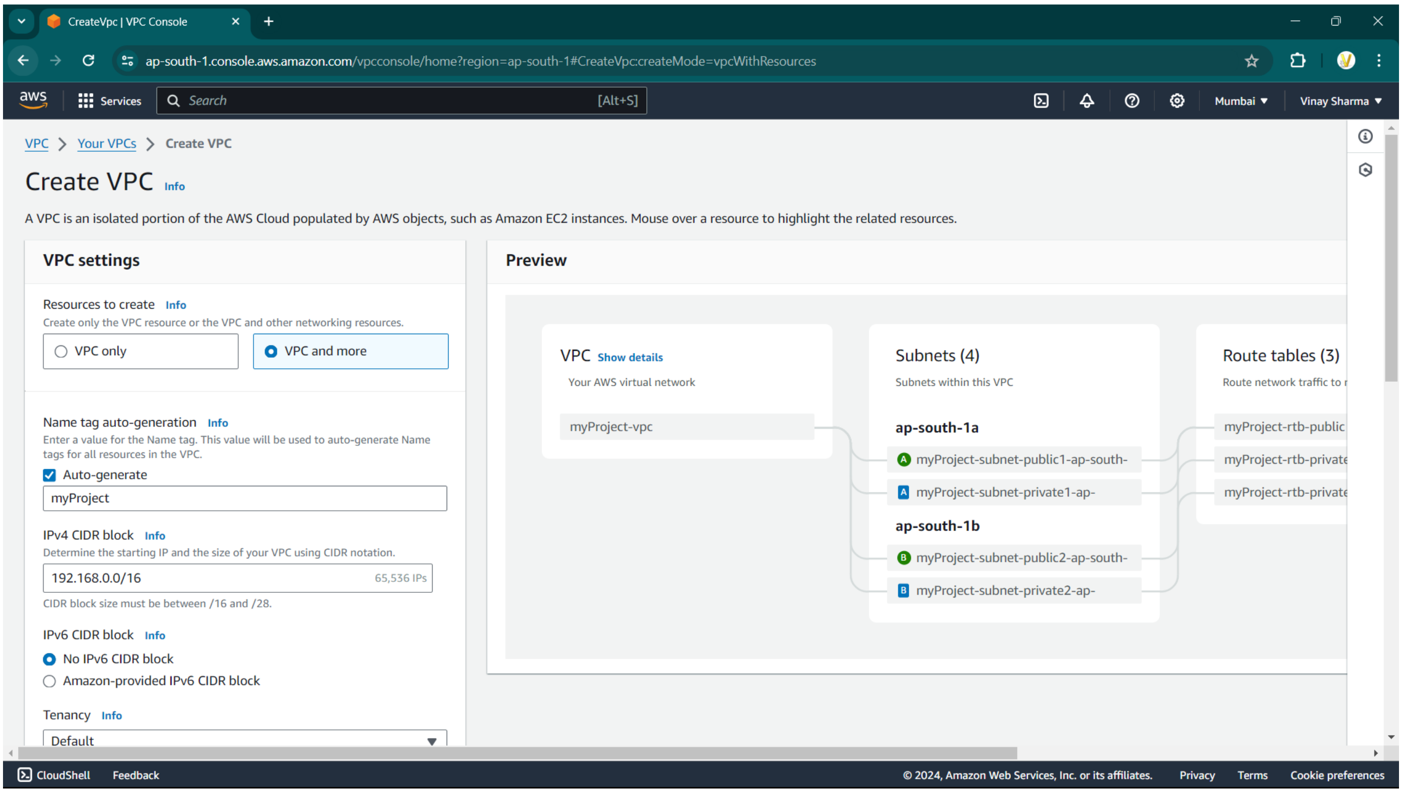
You are using the following Amazon VPC resources

VPCs	Asia Pacific 1	NAT Gateways	Asia Pacific 0
Subnets	Asia Pacific 3	VPC Peering Connections	Asia Pacific 0
Route Tables	Asia Pacific 1	Network ACLs	Asia Pacific 1
Internet Gateways	Asia Pacific 1	Security Groups	Asia Pacific 19
Egress-only Internet Gateways	Asia Pacific 0	Customer Gateways	Asia Pacific 0

AWS Network Manager provides tools and features to help you manage and monitor your network on AWS. Network Manager makes it easier to

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- *Click on create VPC.*


CreateVpc | VPC Console
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateVpc:createMode=vpcWithResources
aws
Services
Search
[Alt+S]
Mumbai ▼
Vinay Sharma ▼
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
myProject

IPv4 CIDR block Info
Determine the starting IP and the size of your VPC using CIDR notation.
192.168.0.0/16 65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
● No IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block

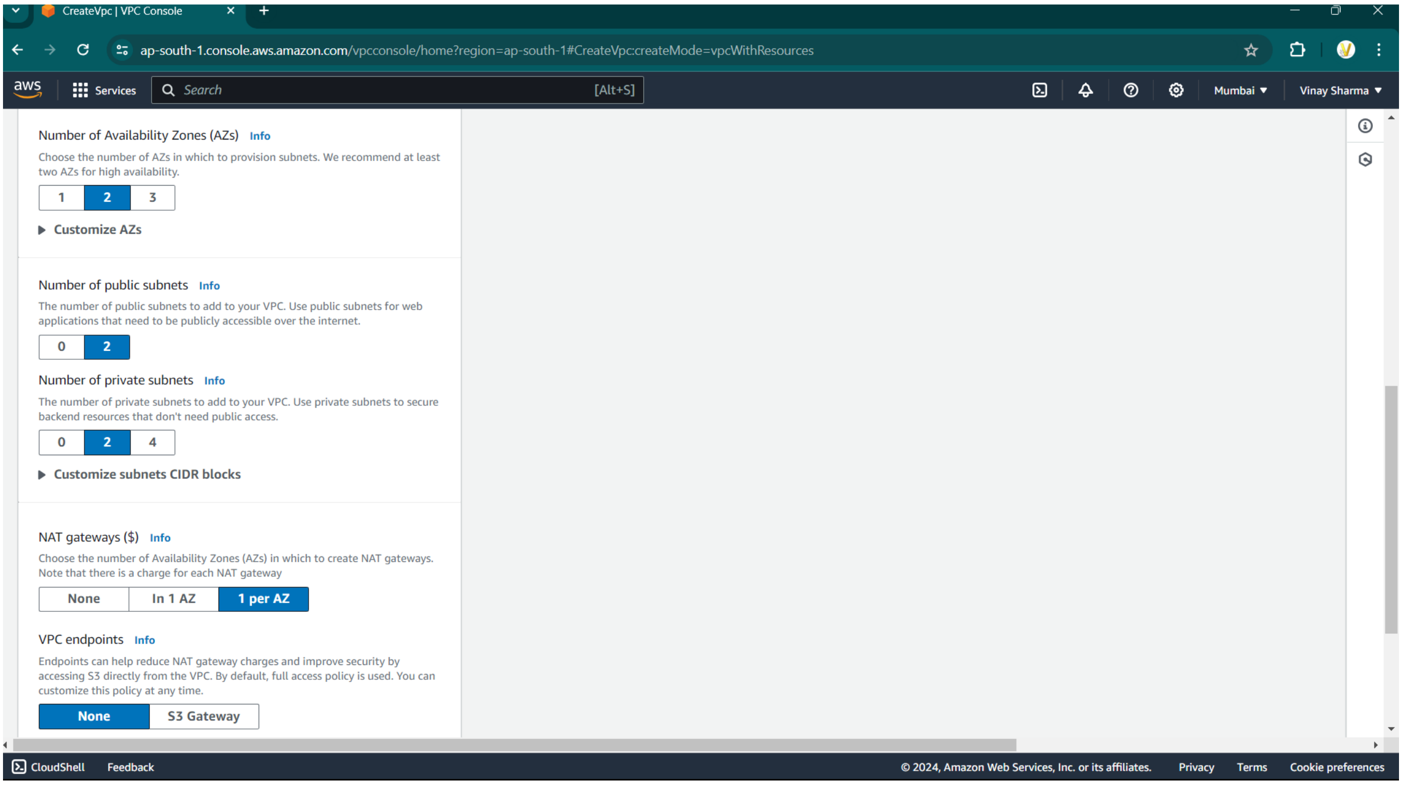
Tenancy Info
Default

Preview

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- *Click VPC and More Because it allows you to add VPC and its component together.*

- *IPv4 CIDR Block is nothing but a size on IP address range.*


CreateVpc | VPC Console
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateVpc:createMode=vpcWithResources
aws
Services
Search
[Alt+S]
Mumbai ▾
Vinay Sharma ▾

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

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

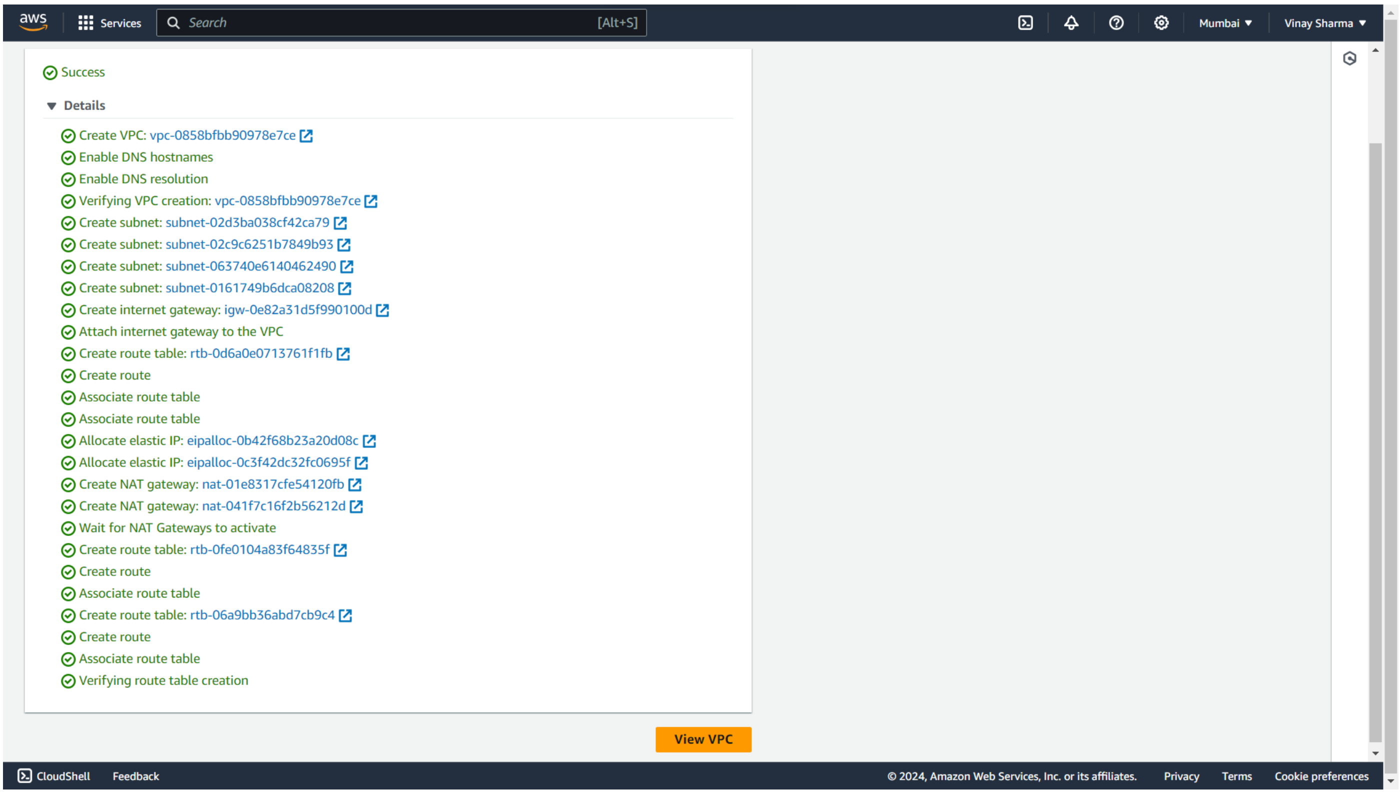
- *I am using Multi-AZ deployment that's why I have choose number of AZ to 2.*
- *In my VPC I want to private and two private subnet in each AZ.*

Public Subnet: has a direct route to an internet gateway. Resources in a public subnet can access the public internet.

Private Subnet: does not have a direct route to an internet gateway.

Internet Gateway: provide two-way public connectivity to applications running in AWS Regions and/or in Local Zones.

NAT Gateway: You can use a NAT gateway so that instances in a private subnet can connect to services outside your VPC but external services cannot initiate a connection with those instances.

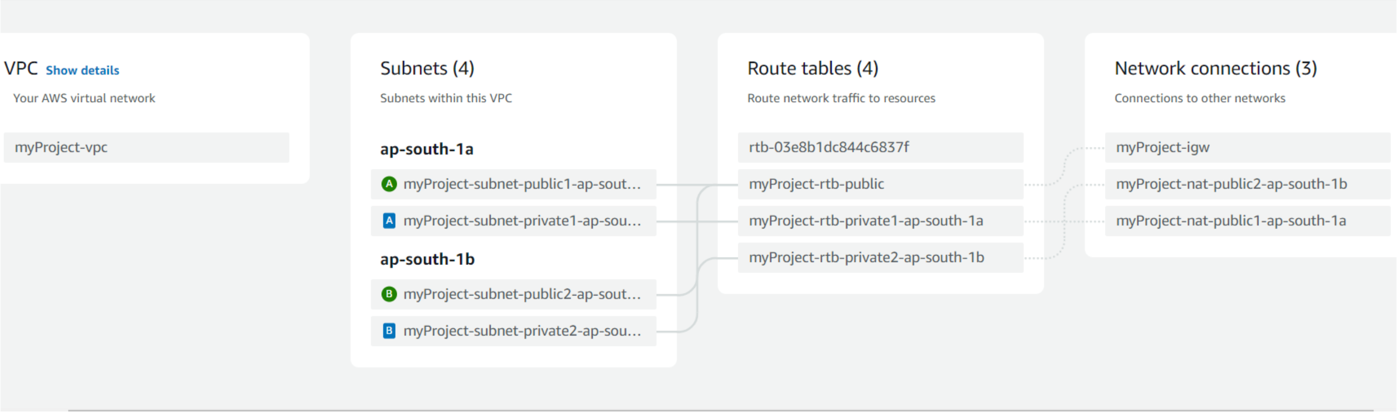


✓ Success

▼ Details

- ✓ Create VPC: vpc-0858bfbb90978e7ce
- ✓ Enable DNS hostnames
- ✓ Enable DNS resolution
- ✓ Verifying VPC creation: vpc-0858bfbb90978e7ce
- ✓ Create subnet: subnet-02d3ba038cf42ca79
- ✓ Create subnet: subnet-02c9c6251b7849b93
- ✓ Create subnet: subnet-063740e6140462490
- ✓ Create subnet: subnet-0161749b6dca08208
- ✓ Create internet gateway: igw-0e82a31d5f990100d
- ✓ Attach internet gateway to the VPC
- ✓ Create route table: rtb-0d6a0e0713761f1fb
- ✓ Create route
- ✓ Associate route table
- ✓ Associate route table
- ✓ Allocate elastic IP: eipalloc-0b42f68b23a20d08c
- ✓ Allocate elastic IP: eipalloc-0c3f42dc32fc0695f
- ✓ Create NAT gateway: nat-01e8317cfe54120fb
- ✓ Create NAT gateway: nat-041f7c16f2b56212d
- ✓ Wait for NAT Gateways to activate
- ✓ Create route table: rtb-0fe0104a83f64835f
- ✓ Create route
- ✓ Associate route table
- ✓ Create route table: rtb-06a9bb36abd7cb9c4
- ✓ Create route
- ✓ Associate route table
- ✓ Verifying route table creation

- *Click on create VPC and it will take some time to create VPC.*


VPC Show details
Your AWS virtual network
myProject-vpc
Subnets (4)
Subnets within this VPC
ap-south-1a
A myProject-subnet-public1-ap-sout…
A myProject-subnet-private1-ap-sou…
ap-south-1b
B myProject-subnet-public2-ap-sout…
B myProject-subnet-private2-ap-sou…
Route tables (4)
Route network traffic to resources
rtb-03e8b1dc844c6837f
myProject-rtb-public
myProject-rtb-private1-ap-south-1a
myProject-rtb-private2-ap-south-1b
Network connections (3)
Connections to other networks
myProject-igw
myProject-nat-public2-ap-south-1b
myProject-nat-public1-ap-south-1a

- *Here is our network design.*

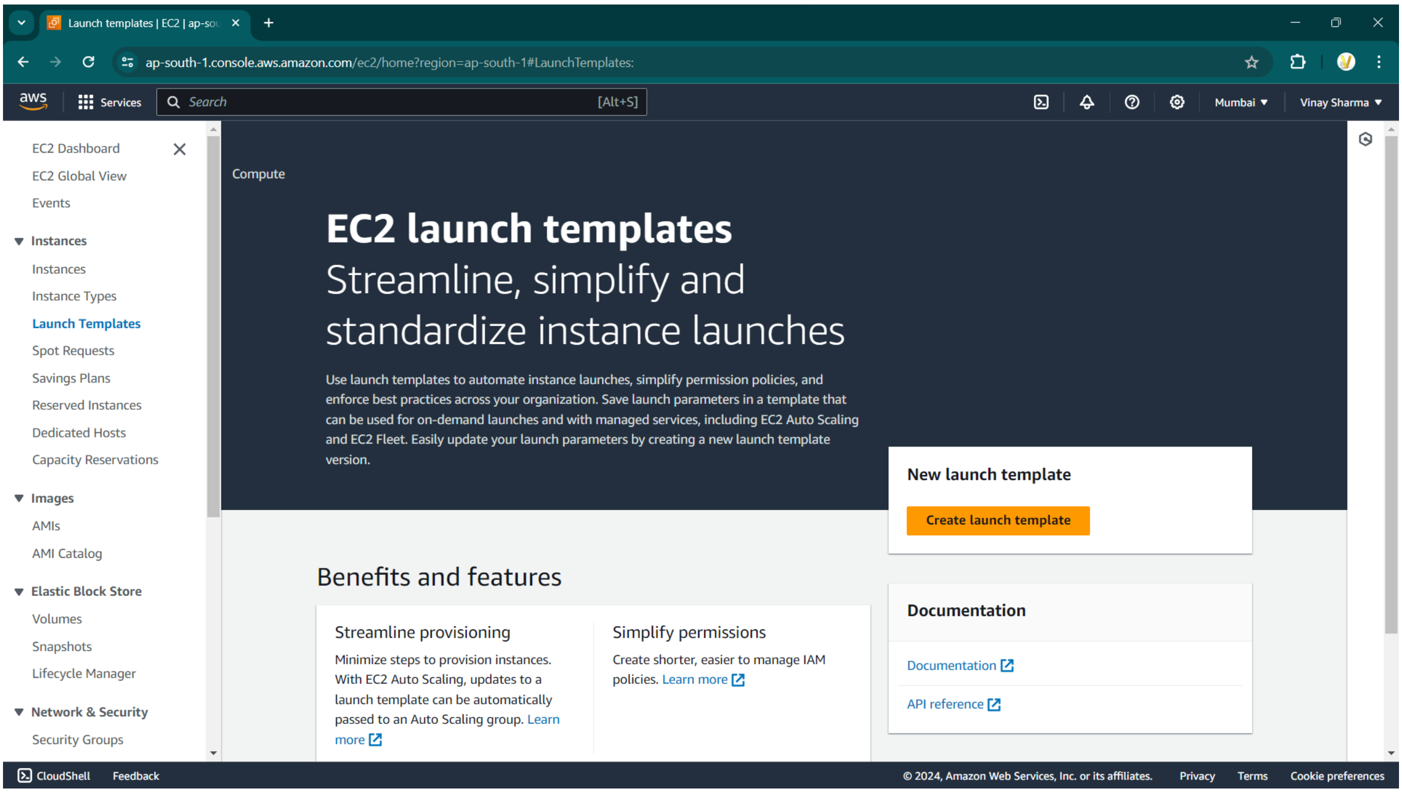
Route Table: contains a set of rules, called routes, that determine where network traffic from your subnet or gateway is directed.

Launch Template: holds instance configuration information. It includes the ID of the Amazon Machine Image (AMI), the instance type, a key pair, security groups, and other parameters used to launch EC2 instances.

Auto Scaling Group: contains a collection of EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management.

Application Load Balancer: operates at the request level (layer 7), routing traffic to targets (EC2 instances, containers, IP addresses, and Lambda functions) based on the content of the request.

Let's Create Launch Template:



EC2 launch templates
Streamline, simplify and standardize instance launches

Use launch templates to automate instance launches, simplify permission policies, and enforce best practices across your organization. Save launch parameters in a template that can be used for on-demand launches and with managed services, including EC2 Auto Scaling and EC2 Fleet. Easily update your launch parameters by creating a new launch template version.

New launch template

Create launch template

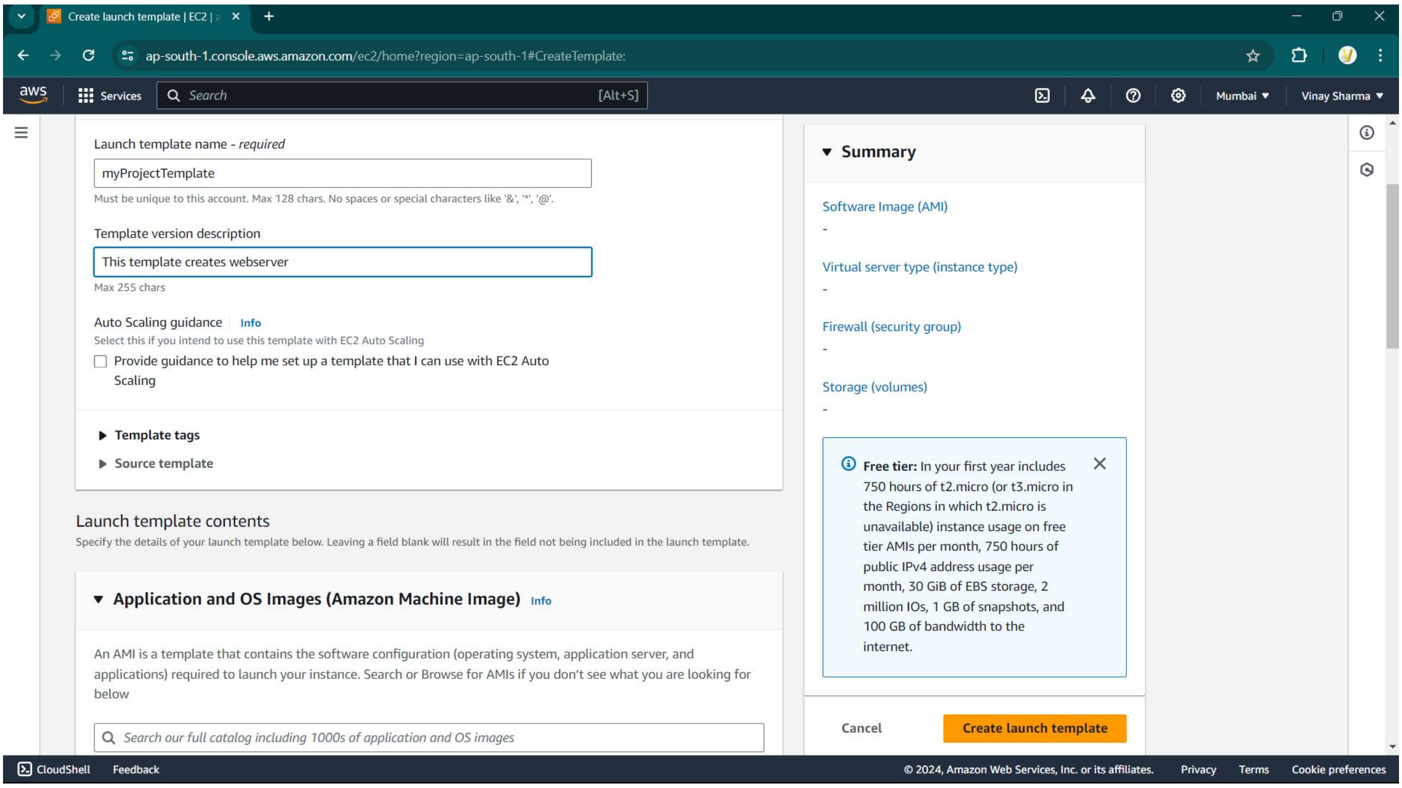
Benefits and features

Streamline provisioning
Minimize steps to provision instances. With EC2 Auto Scaling, updates to a launch template can be automatically passed to an Auto Scaling group. Learn more

Simplify permissions
Create shorter, easier to manage IAM policies. Learn more

Documentation
Documentation
API reference

- *Click on create launch template.*



Launch template name - *required*

myProjectTemplate

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

Template version description

This template creates webserver

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) 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)
-

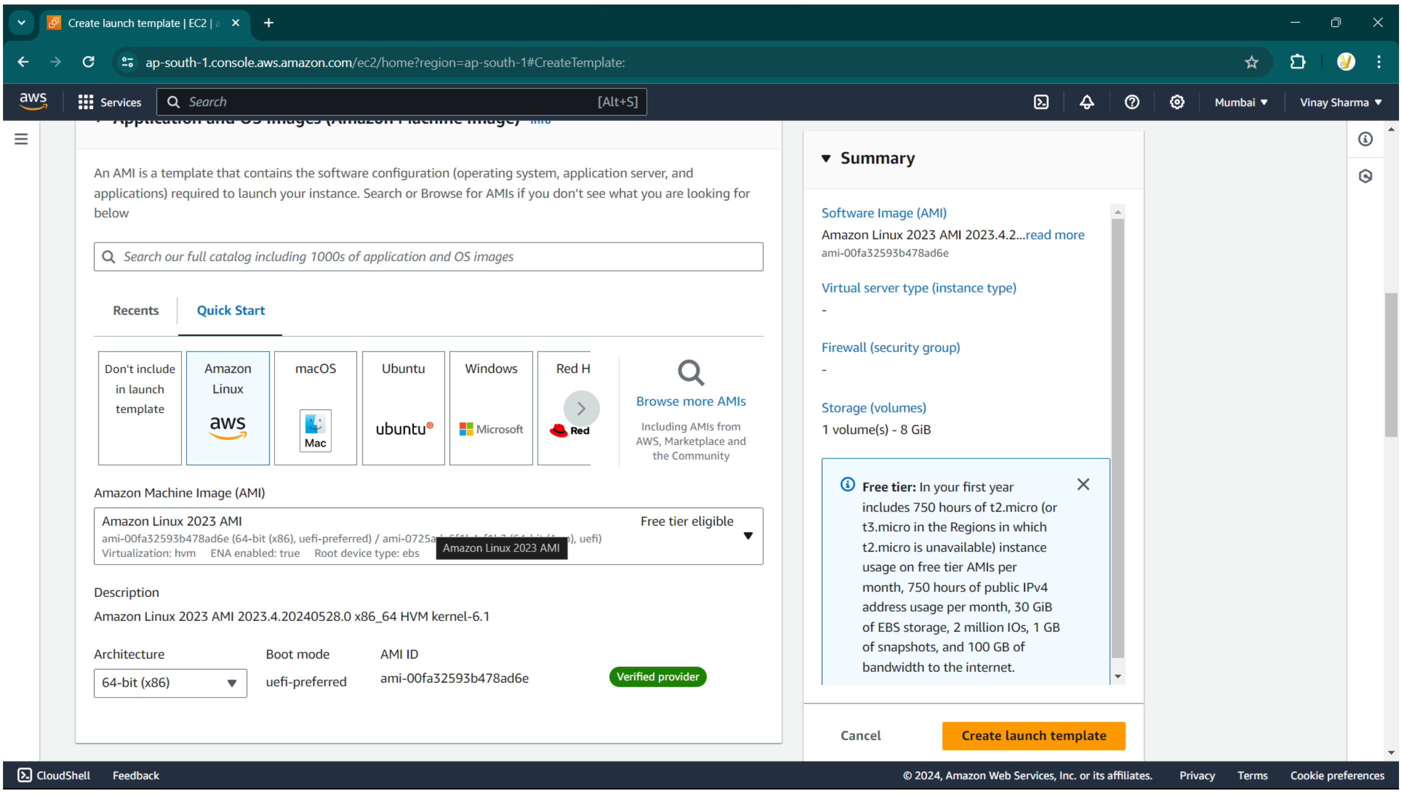
Virtual server type (instance type)
-

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.

Create launch template | EC2 |
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate:
aws
Services
Search
[Alt+S]
Mumbai ▼
Vinay Sharma ▼
Cancel
Create launch template
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



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

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI — Free tier eligible
ami-00fa32593b478ad6e (64-bit (x86), uefi-preferred) / ami-0725a Amazon Linux 2023 AMI), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 AMI 2023.4.20240528.0 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	
64-bit (x86)	uefi-preferred	ami-00fa32593b478ad6e	Verified provider

▼ Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-00fa32593b478ad6e

Virtual server type (instance type)
-

Firewall (security group)
-

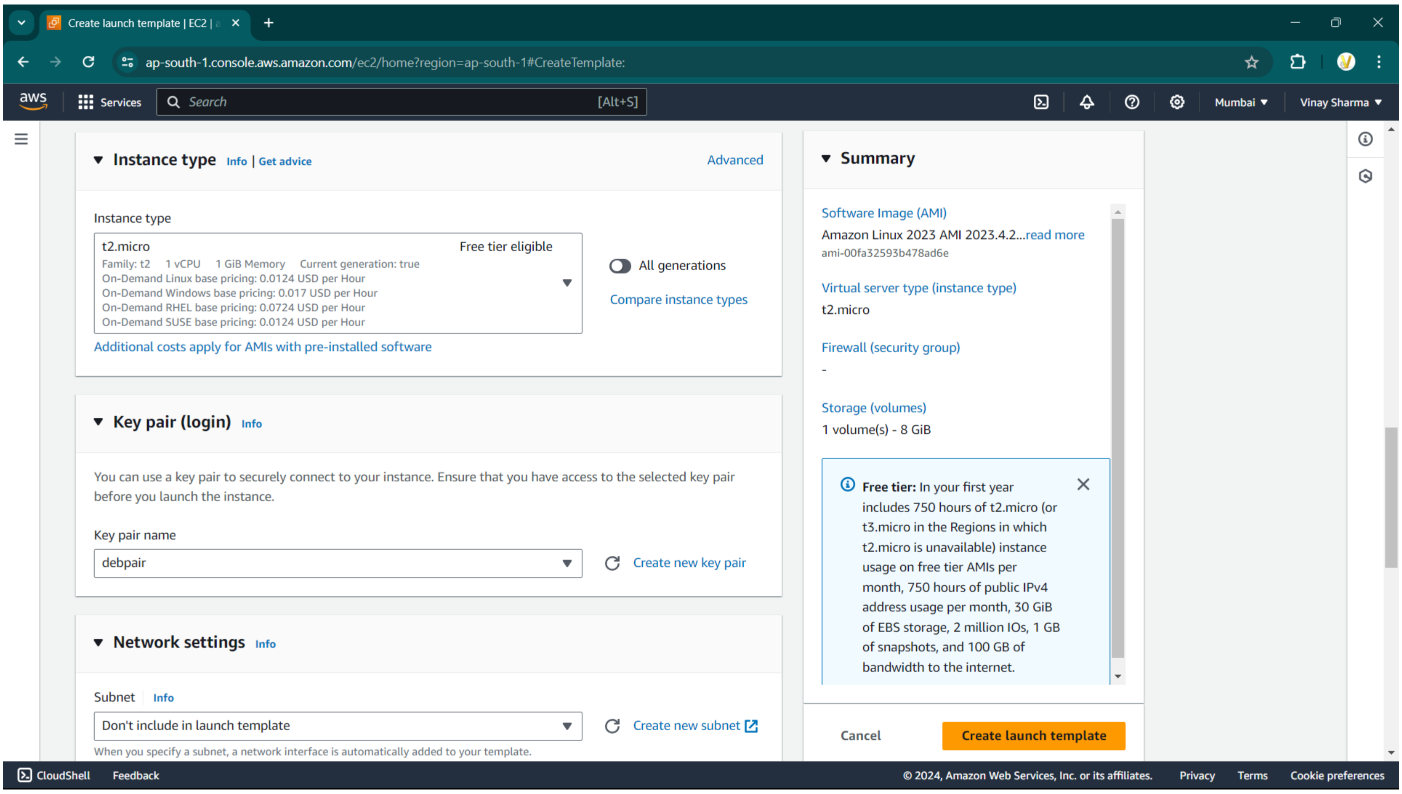
Storage (volumes)
1 volume(s) - 8 GiB

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.

Create launch template | EC2 |
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate:
aws
Services
Search
[Alt+S]
Mumbai ▼
Vinay Sharma ▼
Recents
Quick Start
Don't include in launch template
Amazon Linux
aws
macOS
Mac
Ubuntu
ubuntu
Windows
Microsoft
Red H
Red
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Cancel
Create launch template
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

- *Here I am using Amazon Linux 2 as an AMI.*

Amazon Machine Image(AMI): template that contains a software configuration (for example, an operating system, an application server, and applications). From an AMI, you launch an instance.



Create launch template | EC2 | a
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate:
aws
Services
Search
[Alt+S]
Mumbai ▾
Vinay Sharma ▾

▾ 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 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

Additional costs apply for AMIs with pre-installed software

All generations

Compare instance types

▾ 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: debpair

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.

▾ Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-00fa32593b478ad6e

Virtual server type (instance type)
t2.micro

Firewall (security group)
-

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

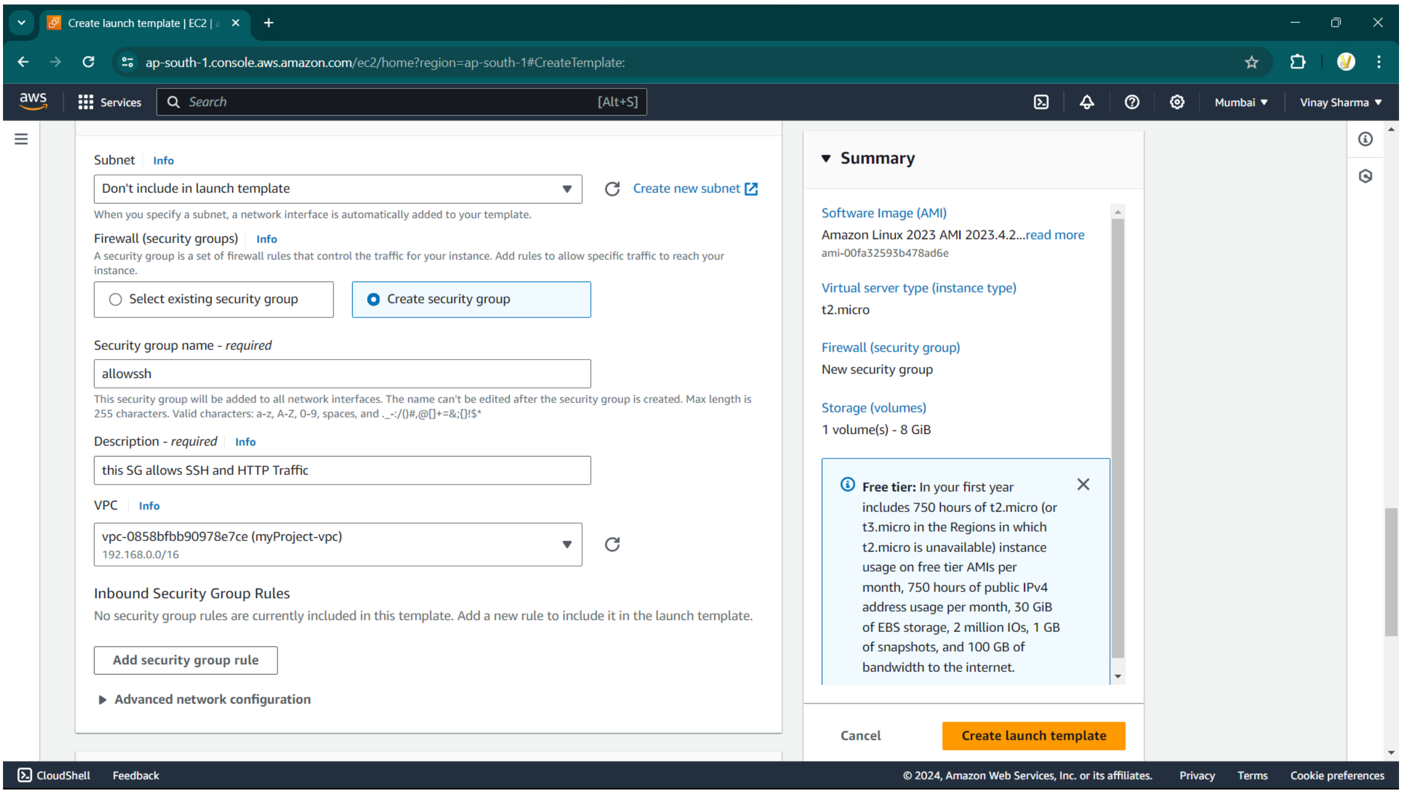
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to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

Cancel | Create launch template

- *Here I am using t2.micro as an instance type which is free one.*

Instance Type: where you specify the hardware of the host computer used for your instance

Key Pair: combination of public and private key that you can use to securely login on your instance.


Create launch template | EC2 |
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate:
aws
Services
Search
[Alt+S]
Mumbai ▾
Vinay Sharma ▾

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 group name - *required*

allowssh

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* Info

this SG allows SSH and HTTP Traffic

VPC Info

vpc-0858bfbb90978e7ce (myProject-vpc)
192.168.0.0/16

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.

Add security group rule

▸ Advanced network configuration

▾ Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-00fa32593b478ad6e

Virtual server type (instance type)
t2.micro

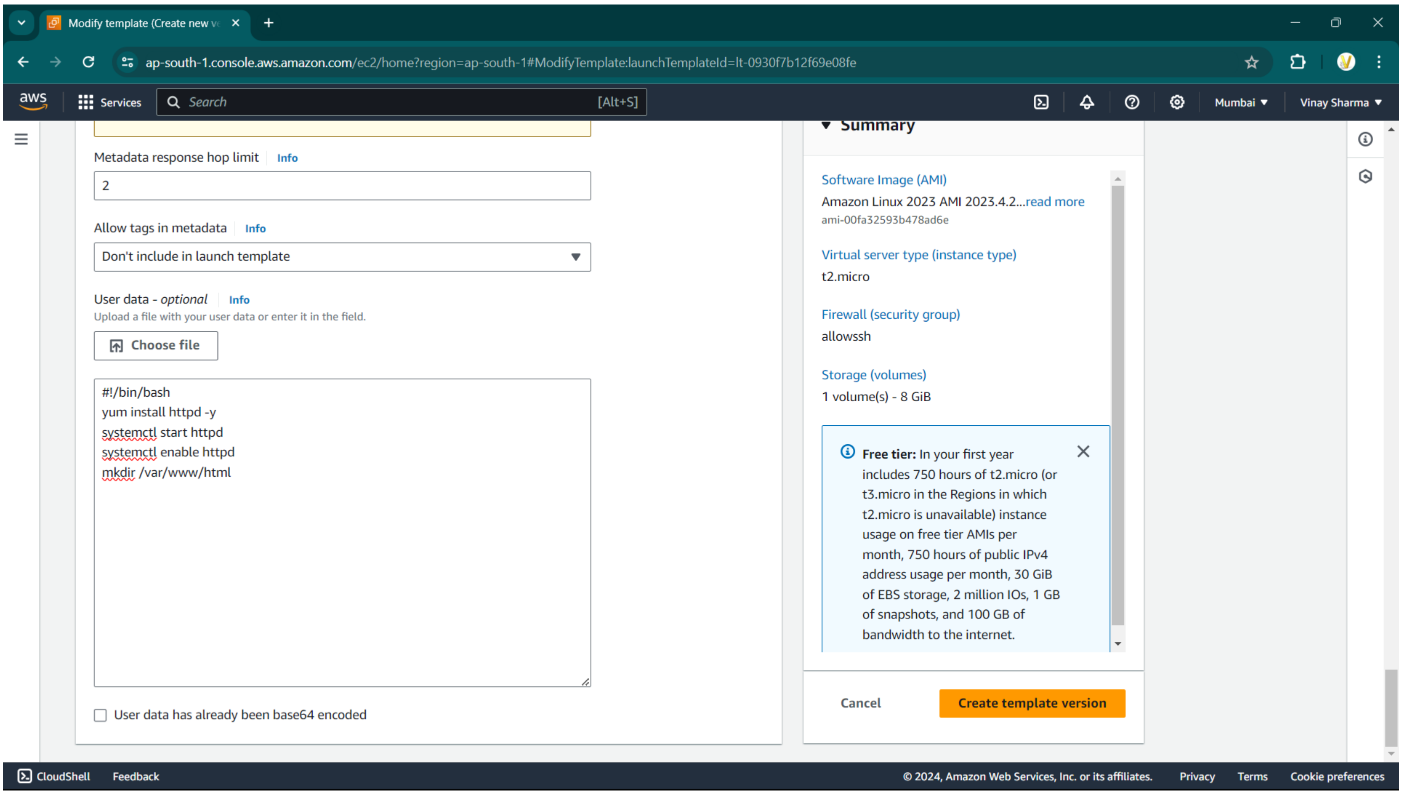
Firewall (security group)
New security group

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

- *Here You have to select the Custom VPC that we have created.*


Modify template (Create new v
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ModifyTemplate:launchTemplateId=lt-0930f7b12f69e08fe
aws
Services
Search
[Alt+S]
Mumbai ▾
Vinay Sharma ▾

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
yum install httpd -y
systemctl start httpd
systemctl enable httpd
mkdir /var/www/html
```

□ User data has already been base64 encoded

▾ Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-00fa32593b478ad6e

Virtual server type (instance type)
t2.micro

Firewall (security group)
allowssh

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.

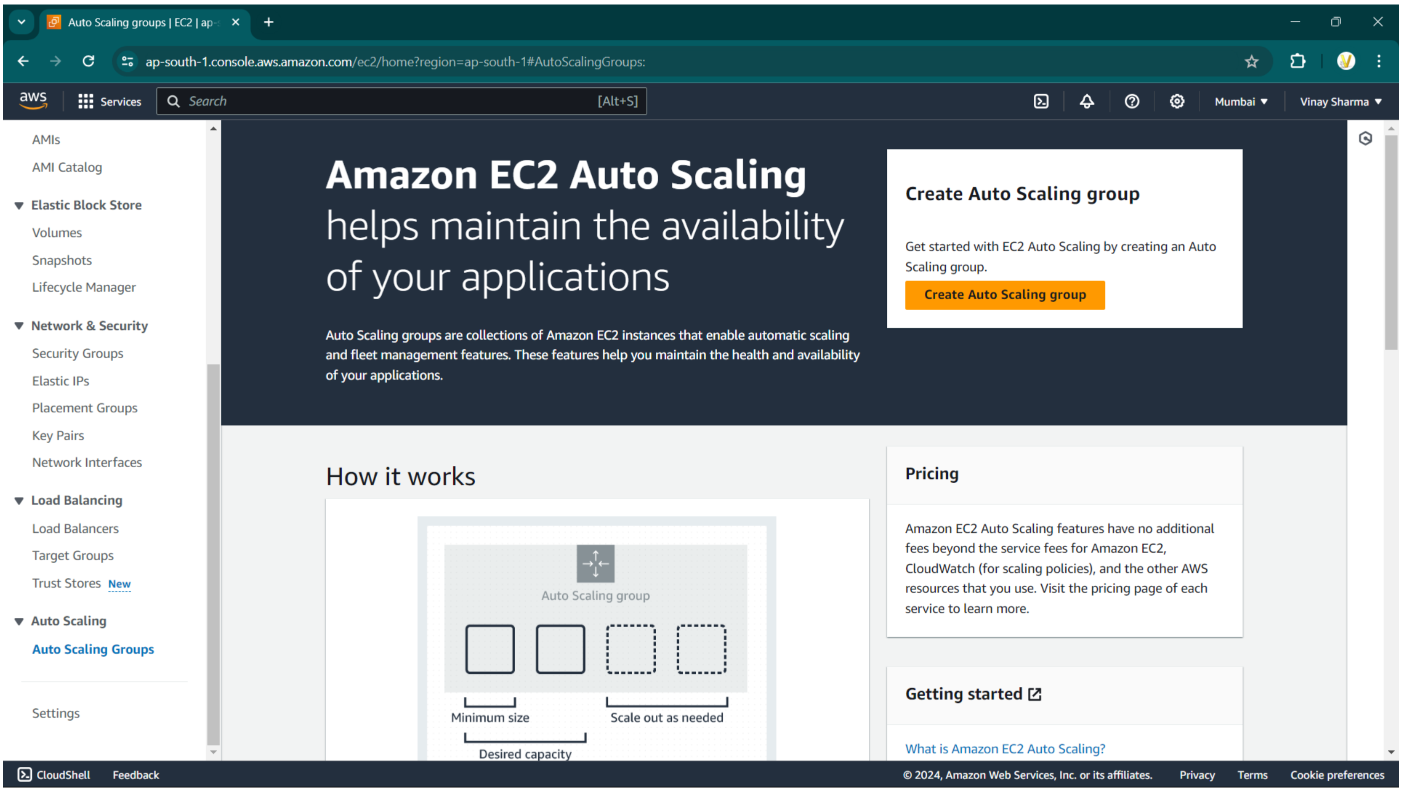
Cancel Create template version

In user data type following command:

```
#!/bin/bash
yum install httpd -y
systemctl start httpd
systemctl enable httpd
mkdir /var/www/html
```

User data: allows user to specify code, task or commands that are going to run at the time of creation of EC2.

Let’s Create Auto Scaling Group & Application Load Balancer:



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

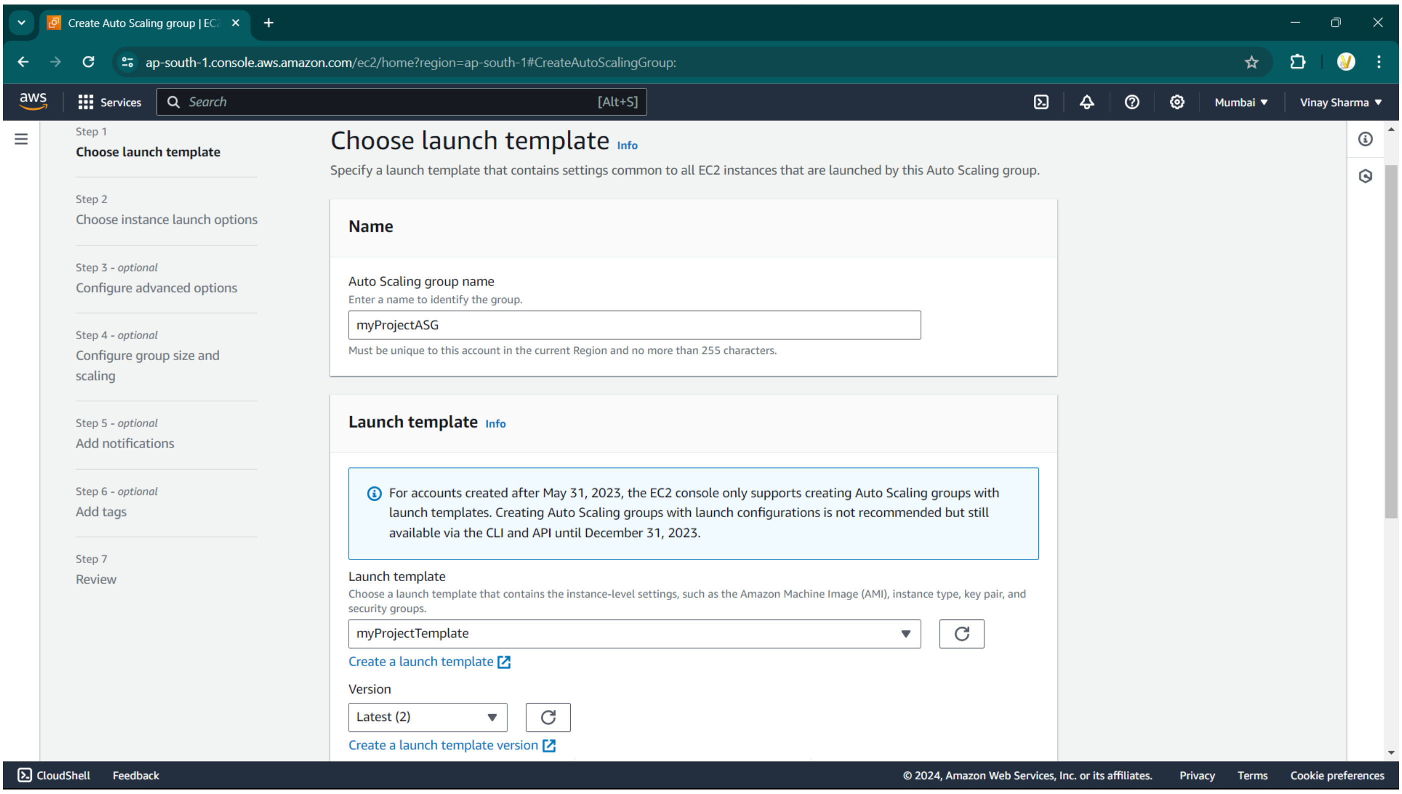
Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Getting started

What is Amazon EC2 Auto Scaling?

© 2024, Amazon Web Services, Inc. or its affiliates.

- *Click on create Auto Scaling Group*



Create Auto Scaling group | EC2
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateAutoScalingGroup:
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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

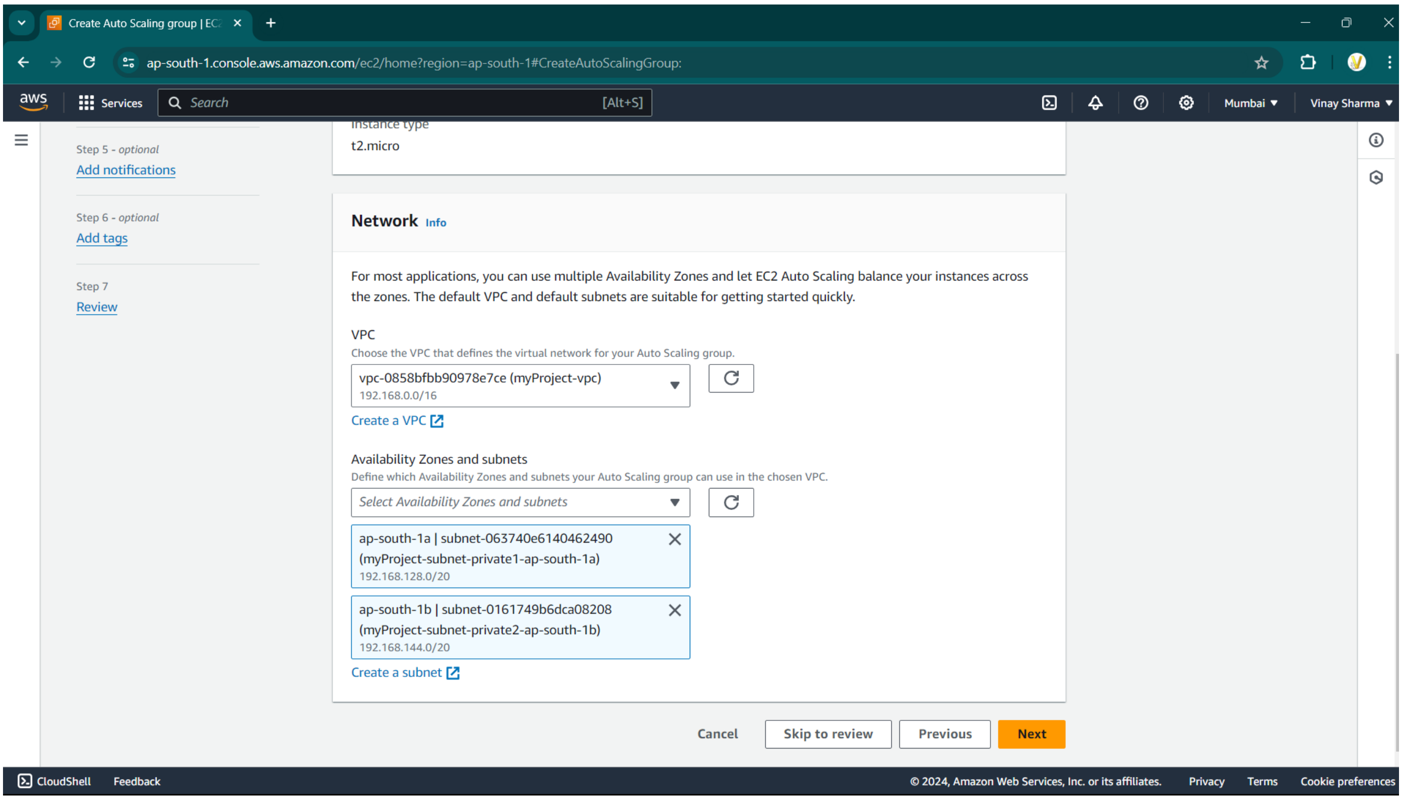
Launch template Info

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.
myProjectTemplate
Create a launch template

Version
Latest (2)
Create a launch template version

- *Here Select Launch Templates that we have created yet.*
- *Click on next.*



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-0858bfbb90978e7ce (myProject-vpc)
192.168.0.0/16
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

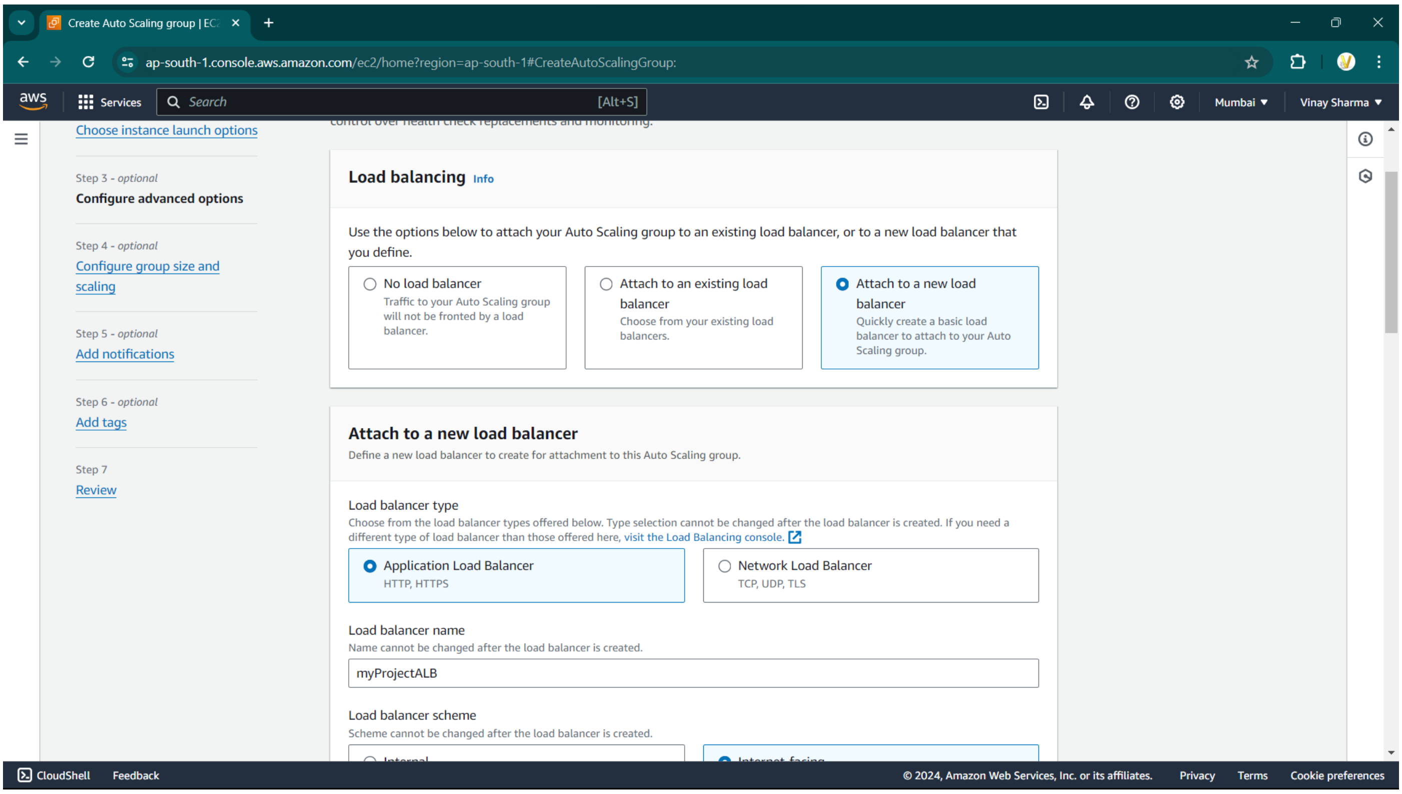
ap-south-1a | subnet-063740e6140462490 (myProject-subnet-private1-ap-south-1a)
192.168.128.0/20

ap-south-1b | subnet-0161749b6dca08208 (myProject-subnet-private2-ap-south-1b)
192.168.144.0/20

Create a subnet

- *In VPC section, select our Custom VPC.*
- *In Subnets section select both of our private subnet, because we want our EC2 servers to run in private subnet.*

- *Click on next.*


Create Auto Scaling group | EC
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateAutoScalingGroup:
aws
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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

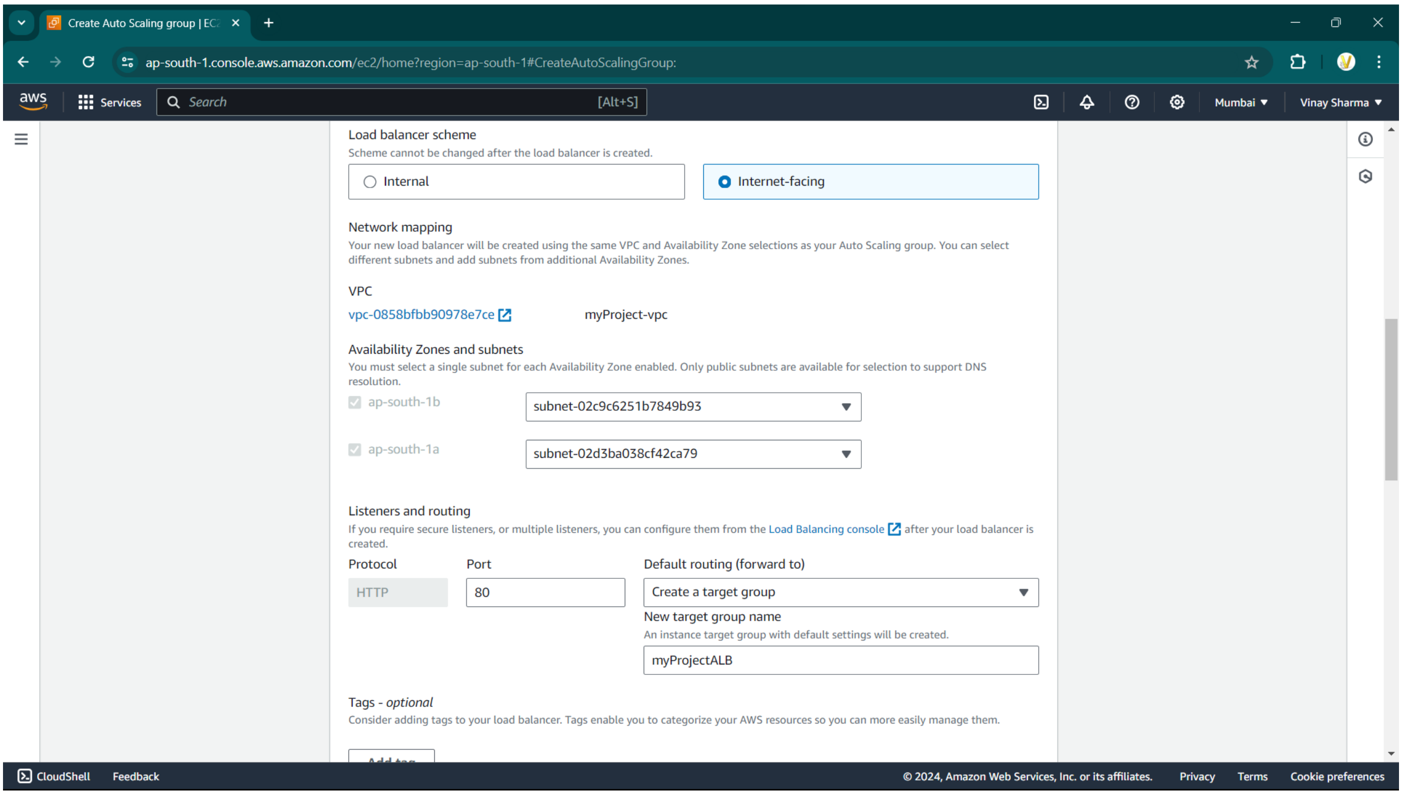
○ Network Load Balancer
TCP, UDP, TLS

Load balancer name
Name cannot be changed after the load balancer is created.
myProjectALB

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

- *In load balancing option, choose attach new load balancer option; this will create and attach load balancer with our ASG.*

- *Choose Application load balancer as a type.*


Create Auto Scaling group | EC
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateAutoScalingGroup:
aws
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
Add tag
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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-0858bfbb90978e7ce myProject-vpc

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-02c9c6251b7849b93

☑ ap-south-1a subnet-02d3ba038cf42ca79

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.

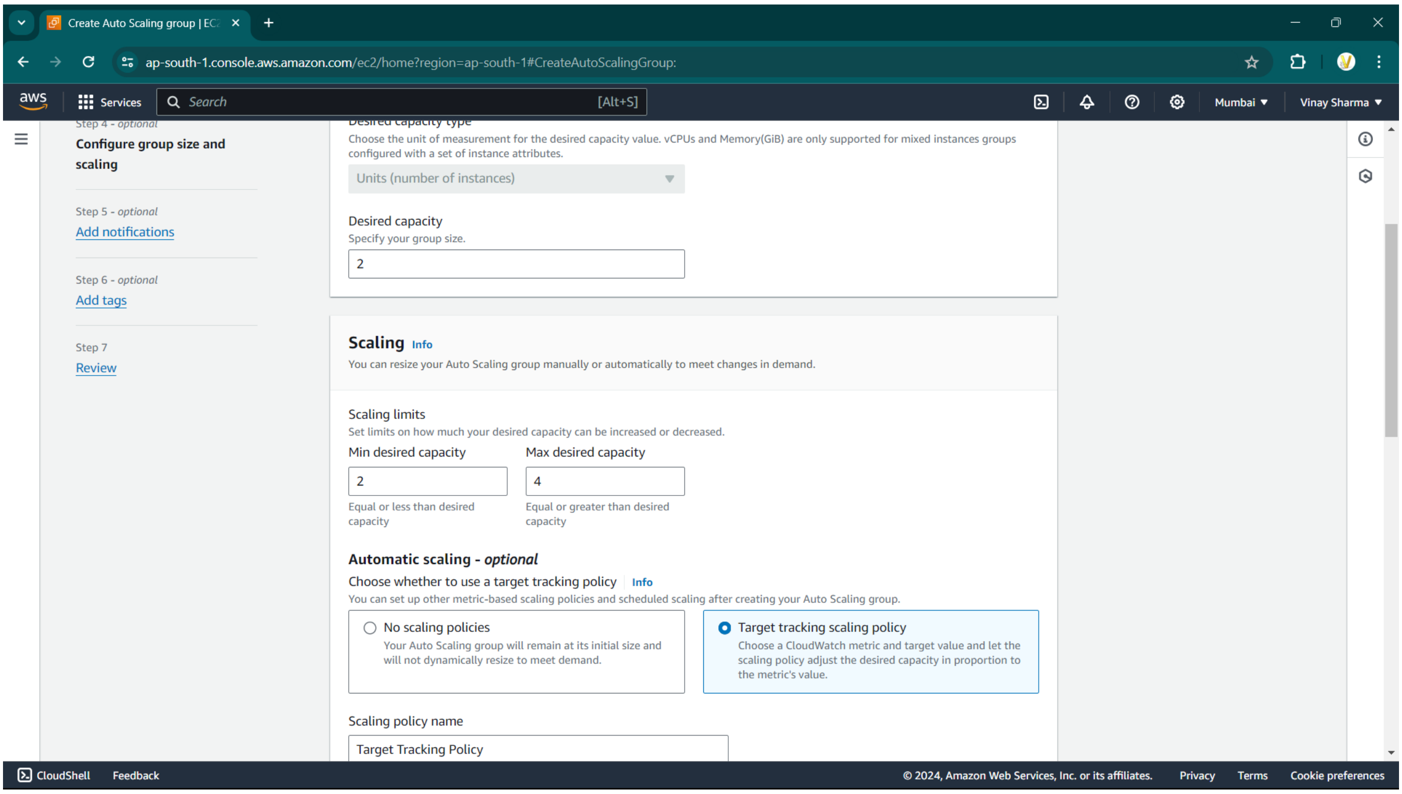
Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group

New target group name
An instance target group with default settings will be created.
myProjectALB

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

- *Make load balancer scheme as a internet-facing; this will make our load balancer to listen from internet.*

- *Choose public subnet; because we want to put our load balancer in public subnet.*



Create Auto Scaling group | EC2
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateAutoScalingGroup:
aws
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 4 - *optional*
Configure group size and scaling

Step 5 - *optional*
Add notifications

Step 6 - *optional*
Add tags

Step 7
Review

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: 4
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.

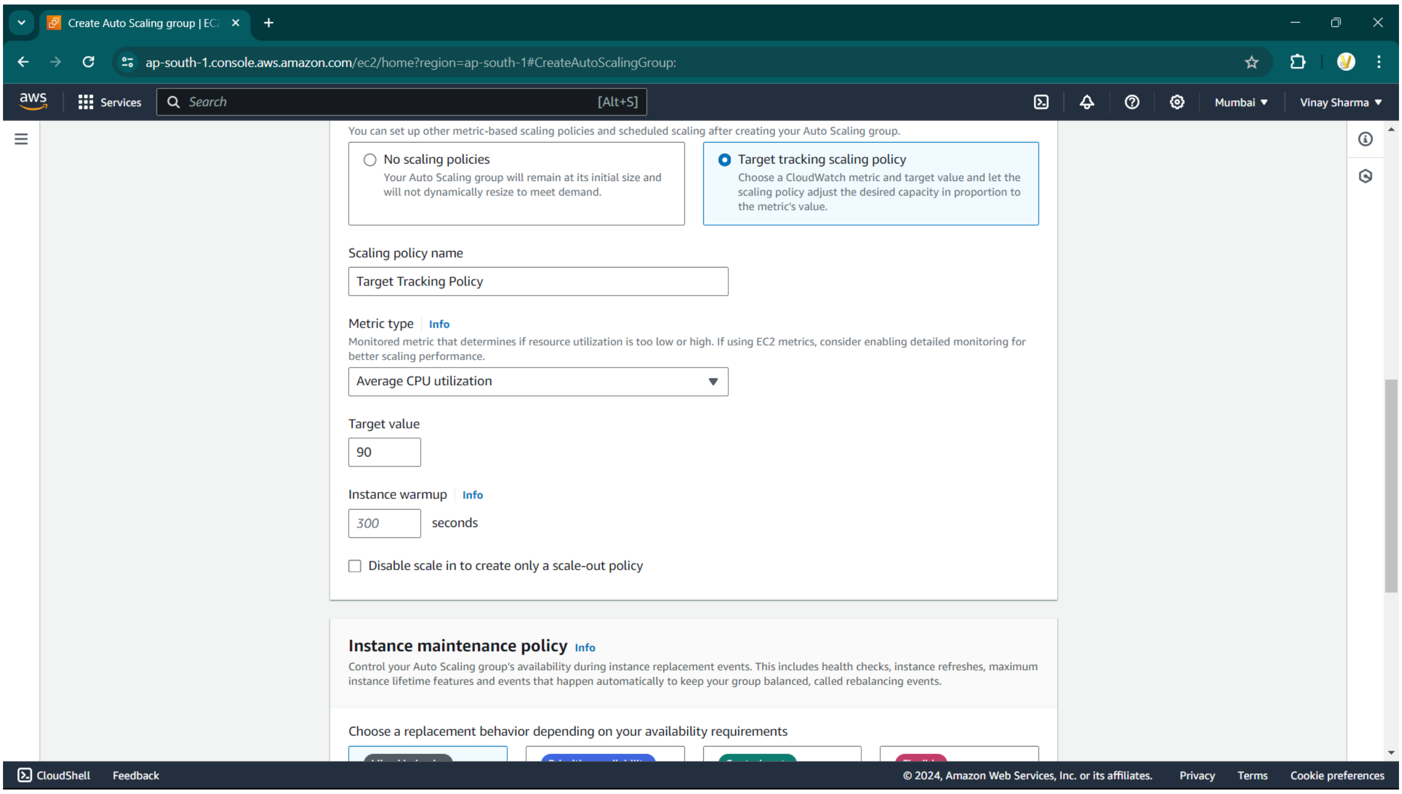
○ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

● 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.

Scaling policy name
Target Tracking Policy

- *In a scaling policy, I want 2 instances each time, that's why I put 2 as a Min desired Capacity.*

- *In Max desired capacity I put 4; when load increases in our2 instances, so it will scale our number of instances to 4.*



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

Metric type Info
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: 90

Instance warmup Info: 300 seconds

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

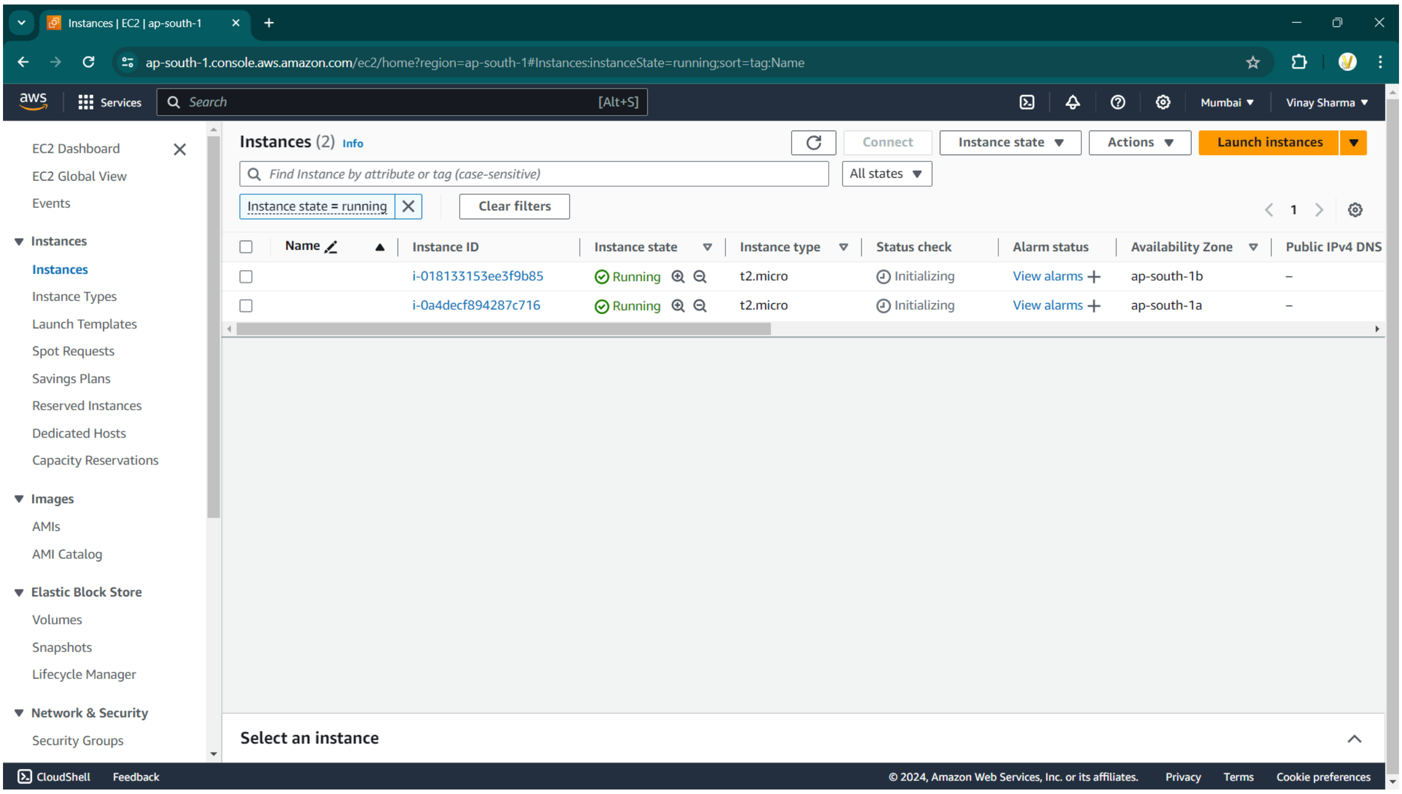
Instance maintenance policy Info
Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Create Auto Scaling group | EC
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateAutoScalingGroup:
aws
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

- *Here I choose metric type as an Average CPU Utilization and target value to 90; that means It will scale EC2 when our existing EC2 instances got more than 90% of CPU utilization.*

- *Now click on create Auto Scaling Group.*



Instances (2) Info

Instance state = running

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
□		i-018133153ee3f9b85	Running	t2.micro	Initializing	View alarms +	ap-south-1b	–
□		i-0a4decf894287c716	Running	t2.micro	Initializing	View alarms +	ap-south-1a	–

Select an instance

- *Here you can see that Auto Scaling Group has automatically created two instances for us; because we have defined desired capacity to 2.*

Bastion Host:

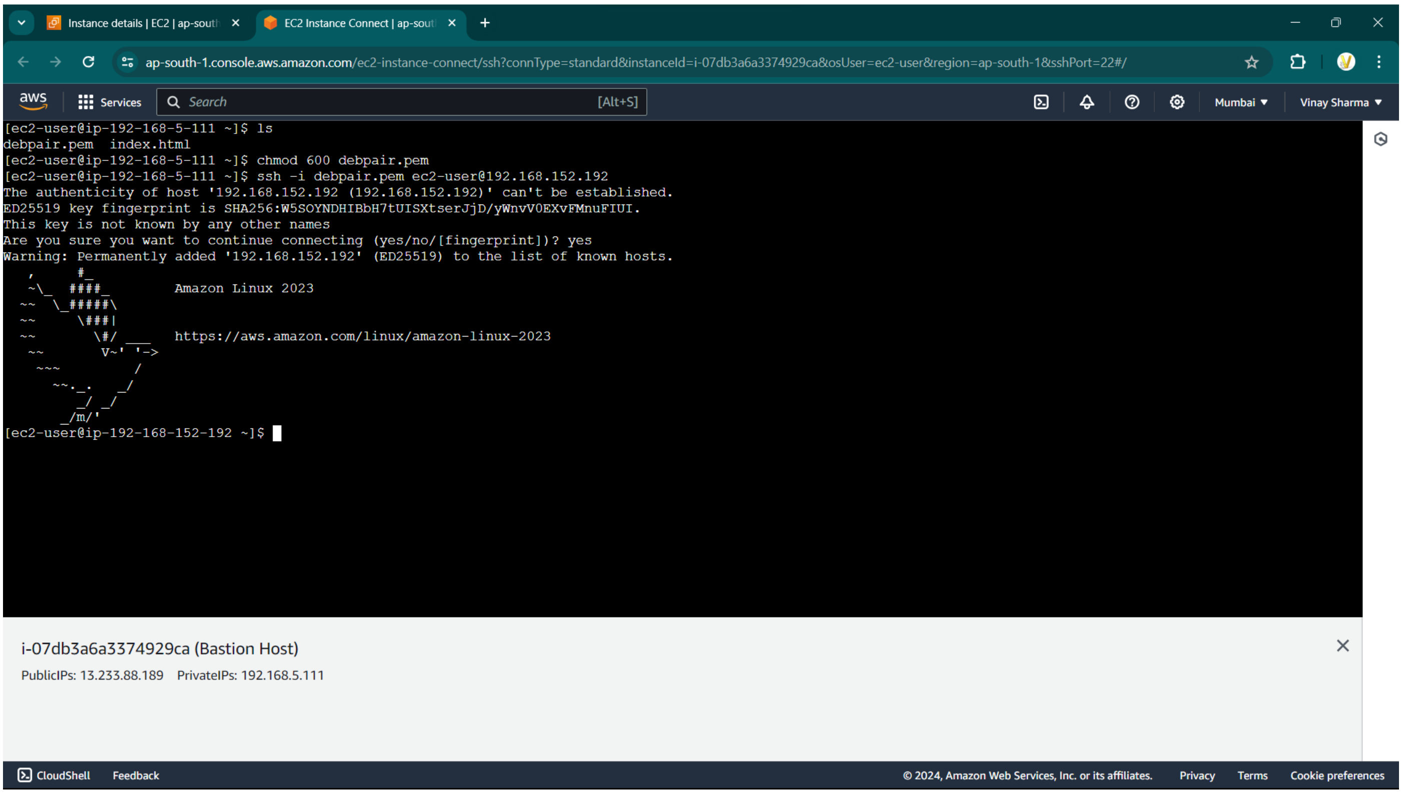
- *Here I have created one more instance that is known as bastion host; I have put this instance in public subnet of my VPC so that I am able to take SSH on it.*

- *I have transferred my index.html file (file that contains my website) & .pem file (that allows me to do SSH to my instances in private subnet) to bastion host using sFTP.*

Bastion host: A bastion host is a server whose purpose is to provide access to a private network from an external network, such as the Internet.

- *Here is the private IP address of both the machine in my private subnet:*

- *192.168.152.192 & 192.168.143.74*



Instance details | EC2 | ap-south
EC2 Instance Connect | ap-south
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-07db3a6a3374929ca&osUser=ec2-user&region=ap-south-1&sshPort=22#/
Services
Search
[Alt+S]
Mumbai
Vinay Sharma

```
[ec2-user@ip-192-168-5-111 ~]$ ls
debpair.pem  index.html
[ec2-user@ip-192-168-5-111 ~]$ chmod 600 debpair.pem
[ec2-user@ip-192-168-5-111 ~]$ ssh -i debpair.pem ec2-user@192.168.152.192
The authenticity of host '192.168.152.192 (192.168.152.192)' can't be established.
ED25519 key fingerprint is SHA256:W5SOYNDHIBbH7tUISXtserJjD/yWnvV0EXvFMnuFIUI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.152.192' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-192-168-152-192 ~]$ 
```

i-07db3a6a3374929ca (Bastion Host)

PublicIPs: 13.233.88.189   PrivateIPs: 192.168.5.111

- *Now I have connected to my bastion host through EC2 instance connect.*

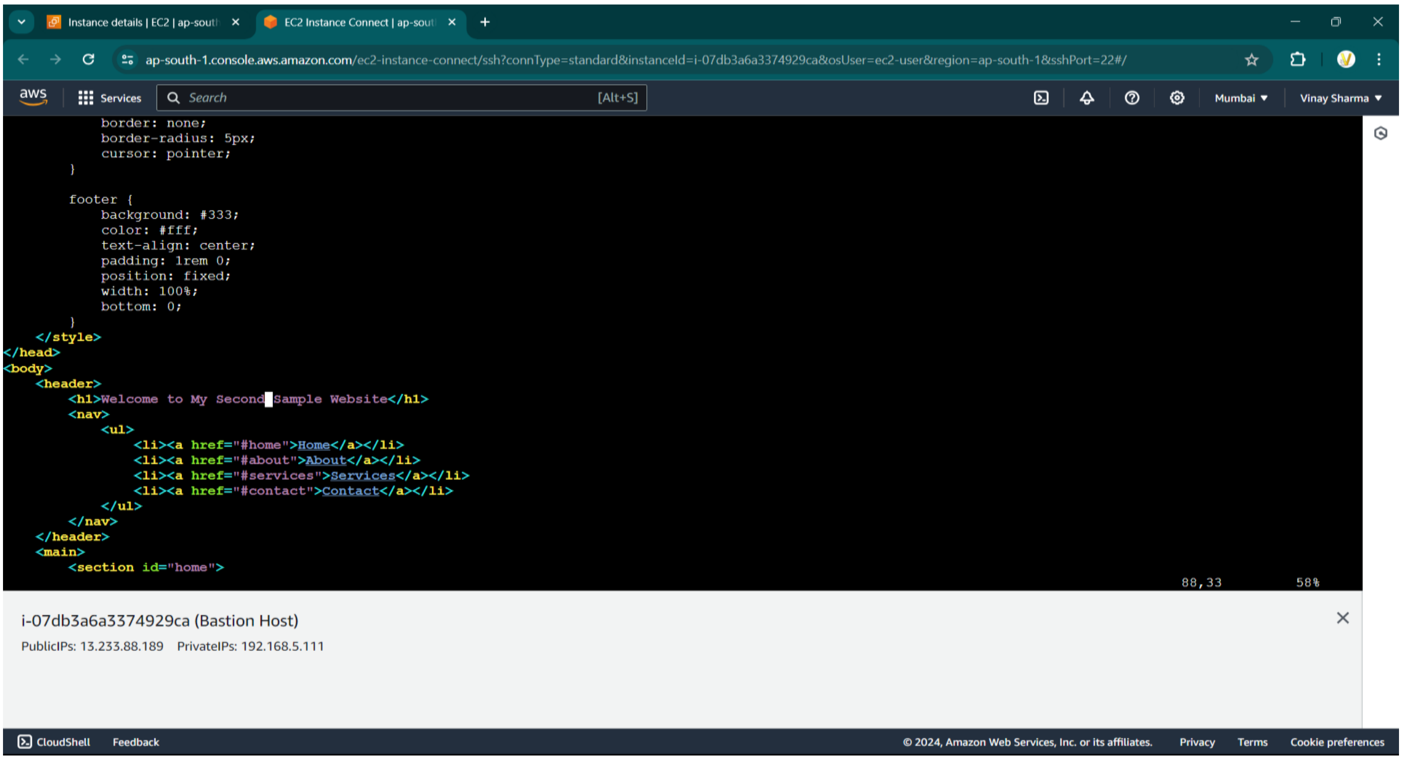
- *Then I uses SCP protocol to transfer index.html file to my both machines inside the private subnet.*

```
scp -i example.pem index.html ec2-user@192.168.152.192:/home/ec2-user
scp -i example.pem index.html ec2-user@192.168.143.74:/home/ec2-user
```

- *Then I uses SSH to connect both of the instances and copy index.html file from /home/ec2-user to /var/www/html/*

```
ssh -i example.pem ec2-user@192.168.152.192
sudo cp index.html /var/www/html
exit
ssh -i example.pem ec2-user@192.168.143.74
```

```
sudo cp index.html /var/www/html
exit
```



Instance details | EC2 | ap-south
EC2 Instance Connect | ap-south
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-07db3a6a3374929ca&osUser=ec2-user&region=ap-south-1&sshPort=22#/
aws
Services
Search
[Alt+S]
Mumbai
Vinay Sharma
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

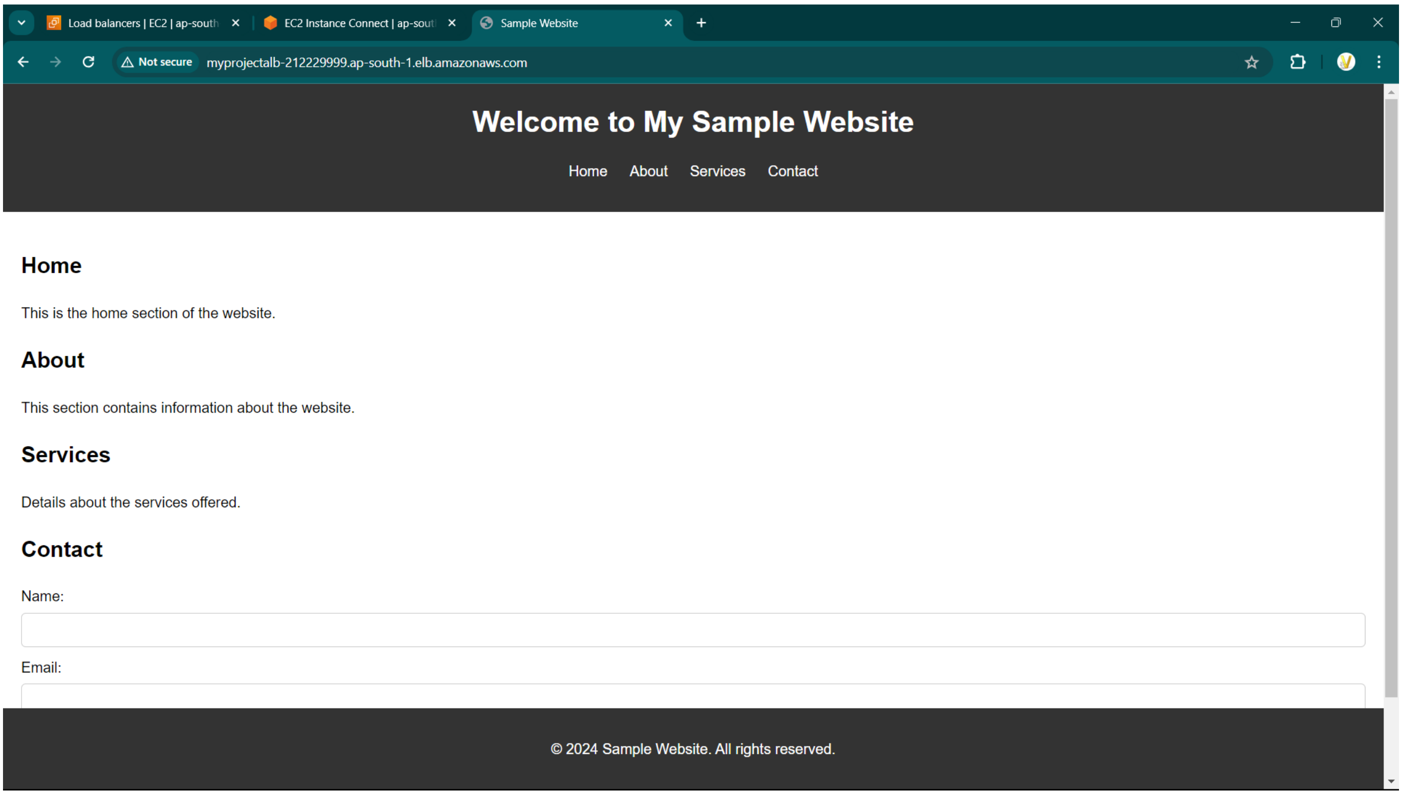
```
            border: none;
            border-radius: 5px;
            cursor: pointer;
        }

        footer {
            background: #333;
            color: #fff;
            text-align: center;
            padding: 1rem 0;
            position: fixed;
            width: 100%;
            bottom: 0;
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome to My Second Sample Website</h1>
        <nav>
            <ul>
                <li><a href="#home">Home</a></li>
                <li><a href="#about">About</a></li>
                <li><a href="#services">Services</a></li>
                <li><a href="#contact">Contact</a></li>
            </ul>
        </nav>
    </header>
    <main>
        <section id="home">
                                                                    88,33         58%
```

i-07db3a6a3374929ca (Bastion Host)

PublicIPs: 13.233.88.189   PrivateIPs: 192.168.5.111

- *In one of the machine I have changed title of my website to ensures load balancing.*



Load balancers | EC2 | ap-south
EC2 Instance Connect | ap-south
Sample Website
Not secure
myprojectalb-212229999.ap-south-1.elb.amazonaws.com

Welcome to My Sample Website

Home About Services Contact

Home

This is the home section of the website.

About

This section contains information about the website.

Services

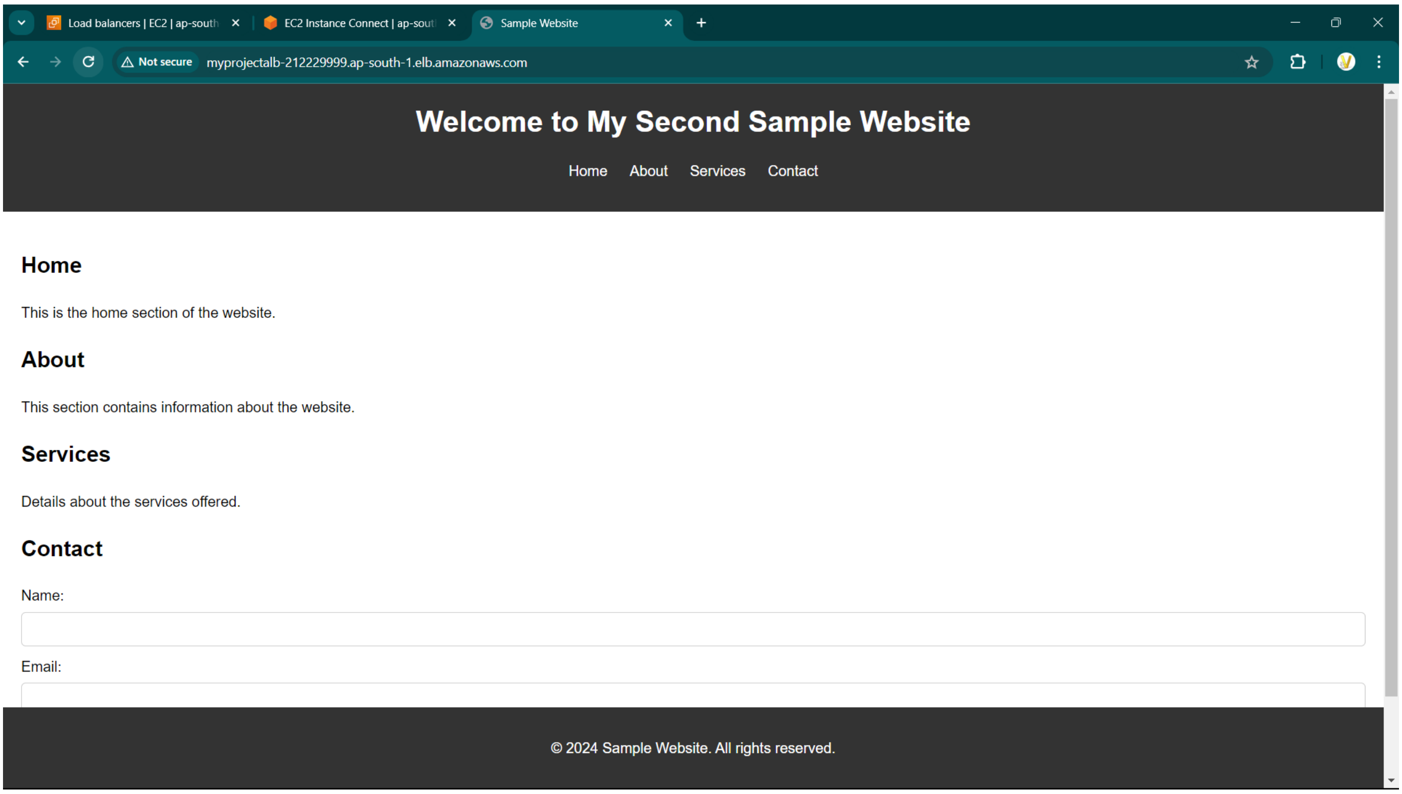
Details about the services offered.

Contact

Name:

Email:

© 2024 Sample Website. All rights reserved.


Load balancers | EC2 | ap-south
EC2 Instance Connect | ap-sout
Sample Website
Not secure
myprojectalb-212229999.ap-south-1.elb.amazonaws.com

Welcome to My Second Sample Website

Home About Services Contact

Home

This is the home section of the website.

About

This section contains information about the website.

Services

Details about the services offered.

Contact

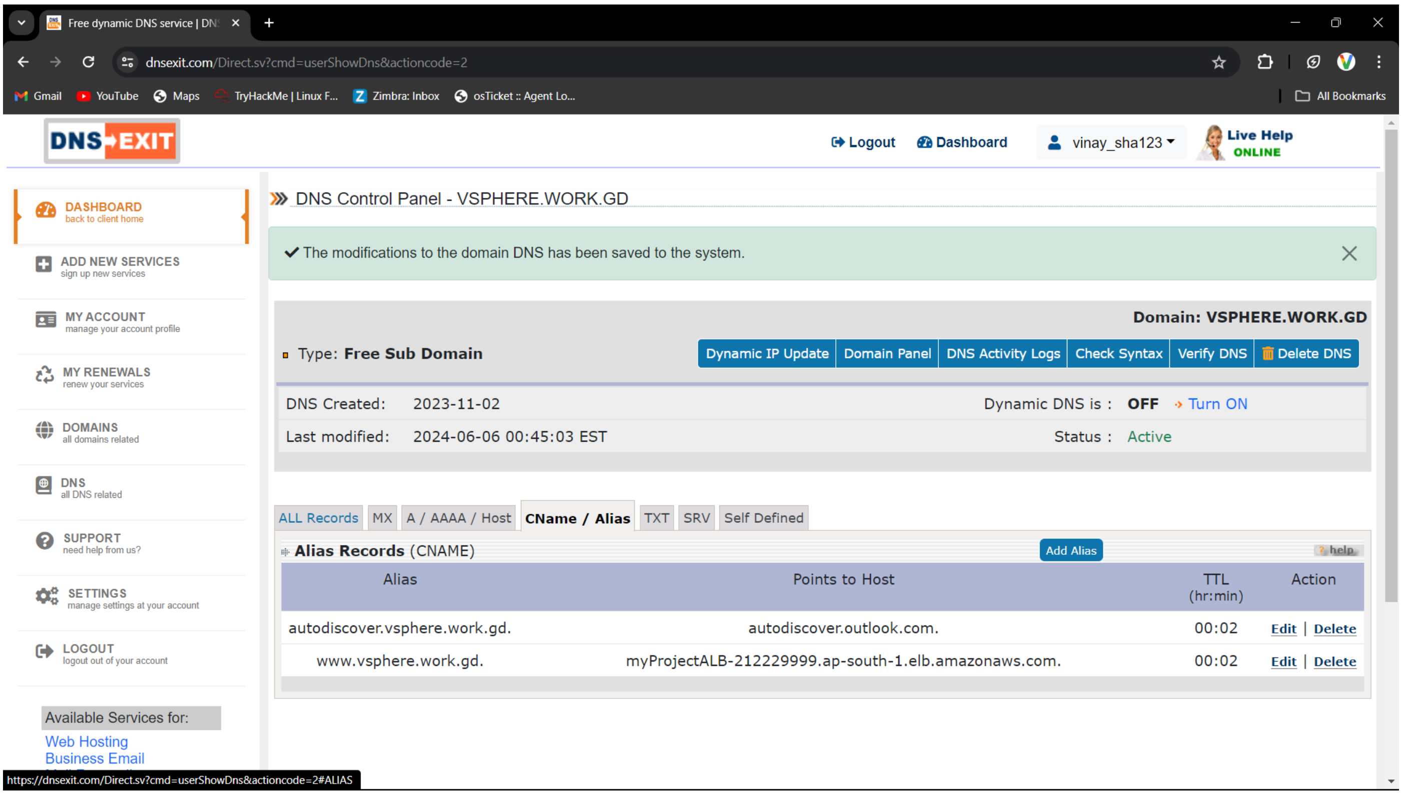
Name:

Email:

© 2024 Sample Website. All rights reserved.

- *Using load balancer's DNS name, my website is got accessed. When I refresh my page, so you can see that my request is got redirected to another instance; that means my load balancer works fine!!!*

Let's Add Entry in DNS:



DNS Control Panel - VSPHERE.WORK.GD

✓ The modifications to the domain DNS has been saved to the system.

Domain: VSPHERE.WORK.GD

Type: **Free Sub Domain**

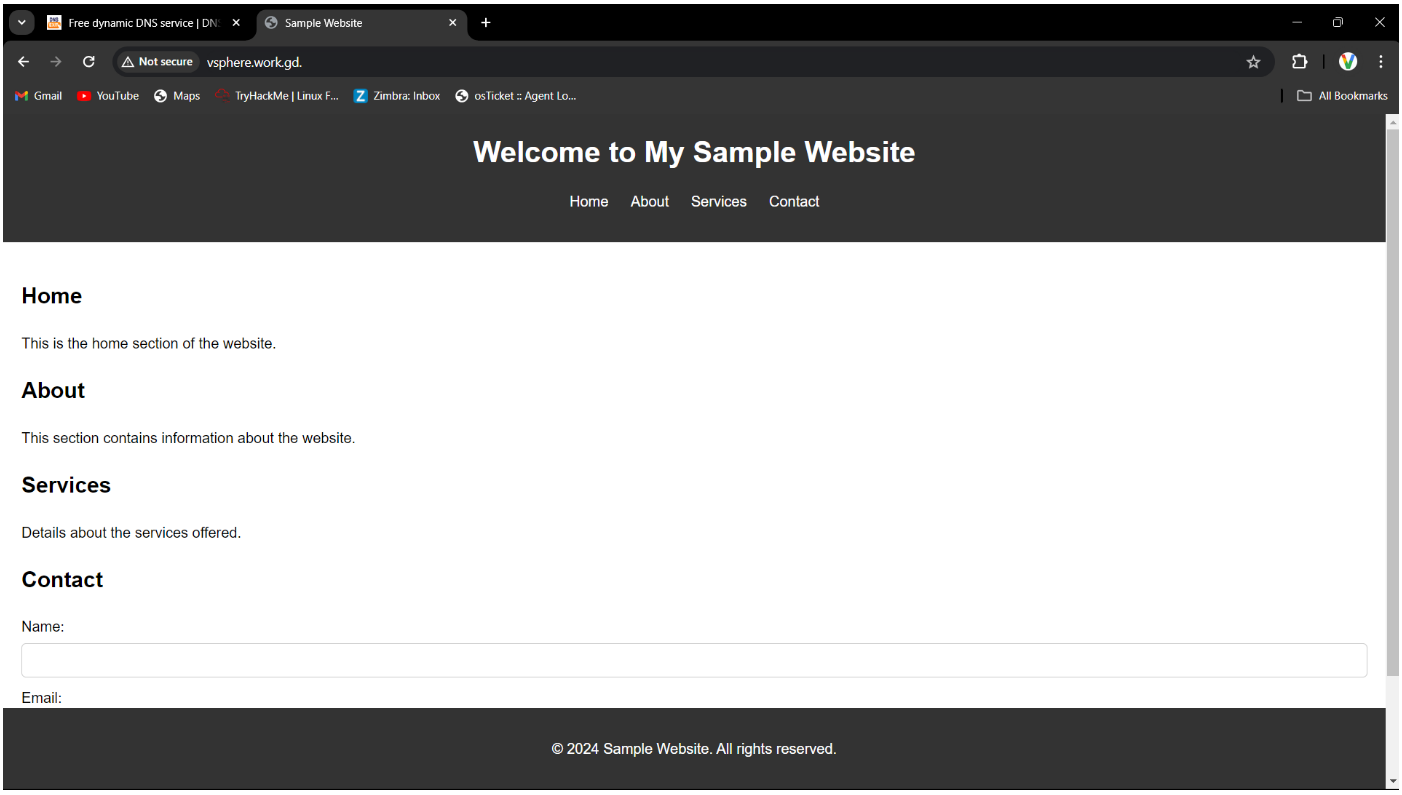
DNS Created: 2023-11-02   Dynamic DNS is : **OFF** Turn ON

Last modified: 2024-06-06 00:45:03 EST   Status : Active

Alias Records (CNAME)

Alias	Points to Host	TTL (hr:min)	Action
autodiscover.vsphere.work.gd.	autodiscover.outlook.com.	00:02	Edit | Delete
www.vsphere.work.gd.	myProjectALB-212229999.ap-south-1.elb.amazonaws.com.	00:02	Edit | Delete

- *I want my website to get accessed using my own domain name “vsphere.work.gd”. so I have created CNAME record that points www.vshpere.work.gd to my load balancer’s DNS name.*



Welcome to My Sample Website

Home About Services Contact

Home

This is the home section of the website.

About

This section contains information about the website.

Services

Details about the services offered.

Contact

Name:

Email:

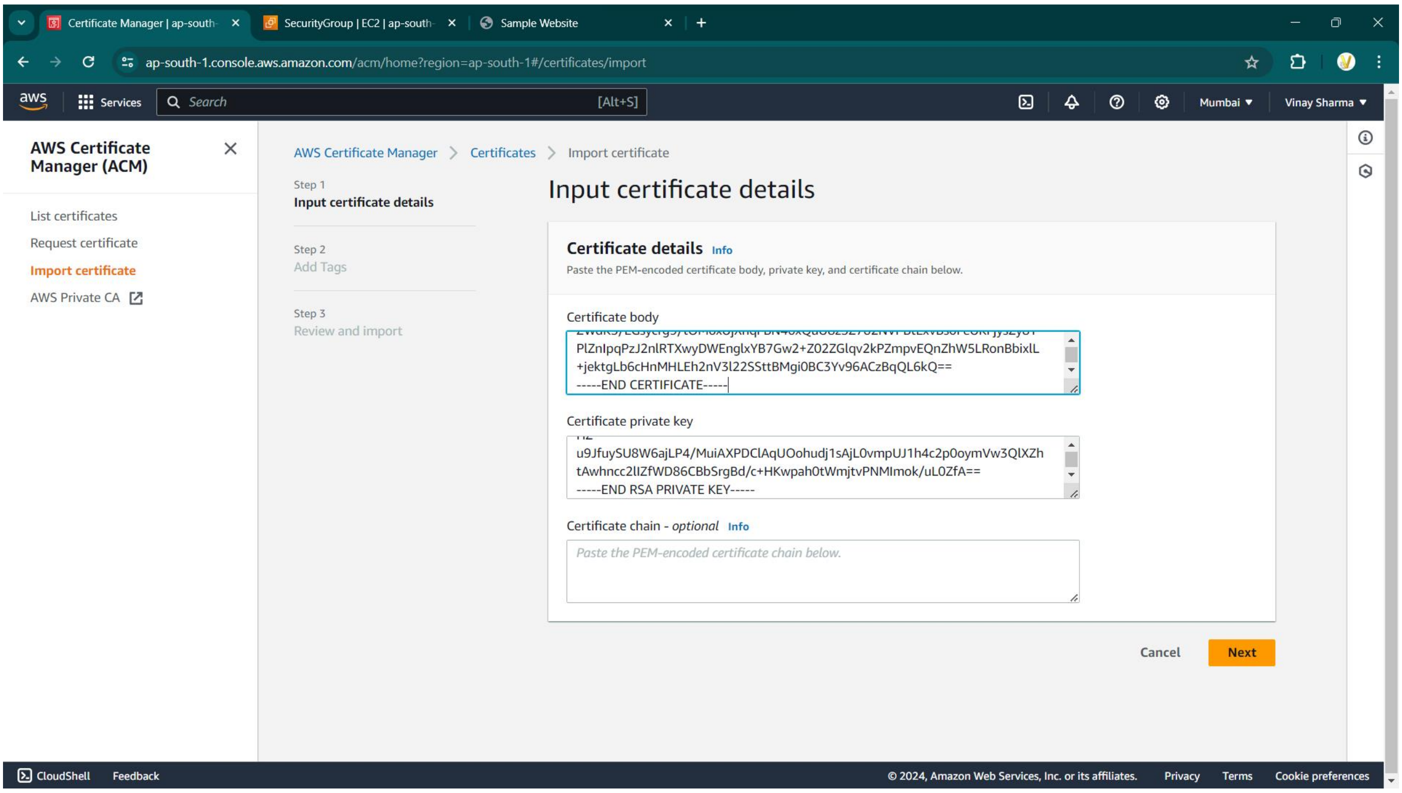
© 2024 Sample Website. All rights reserved.

- *Now you can see that, my website is got accessed through my own domain name.*

Let’s Generate SSL Certificate:

SSL Certificate: digital certificate that authenticates a website’s identity and enables an encrypted connection.

Amazon Certificate Manager: AWS Certificate Manager (ACM) is a service that lets you easily provision, manage, and deploy public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates.


Certificate Manager | ap-south-
SecurityGroup | EC2 | ap-south-
Sample Website
ap-south-1.console.aws.amazon.com/acm/home?region=ap-south-1#/certificates/import
aws
Services
Search
[Alt+S]
Mumbai ▾
Vinay Sharma ▾
AWS Certificate Manager (ACM)
List certificates
Request certificate
Import certificate
AWS Private CA
AWS Certificate Manager > Certificates > Import certificate
Step 1
Input certificate details
Step 2
Add Tags
Step 3
Review and import

Input certificate details

Certificate details Info

Paste the PEM-encoded certificate body, private key, and certificate chain below.

Certificate body

```
PlZnIpqPzJ2nlRTXwyDWEnglxYB7Gw2+Z02ZGlqv2kPZmpvEQnZhW5LRonBbixlL
+jektgLb6cHnMHLEh2nV3l22SSttBMgi0BC3Yv96ACzBqQL6kQ==
-----END CERTIFICATE-----
```

Certificate private key

```
u9JfuySU8W6ajLP4/MuiAXPDClAqUOohudj1sAjL0vmpUJ1h4c2p0oymVw3QlXZh
tAwhncc2lIZfWD86CBbSrgBd/c+HKwpah0tWmjtvPNMImok/uL0ZfA==
-----END RSA PRIVATE KEY-----
```

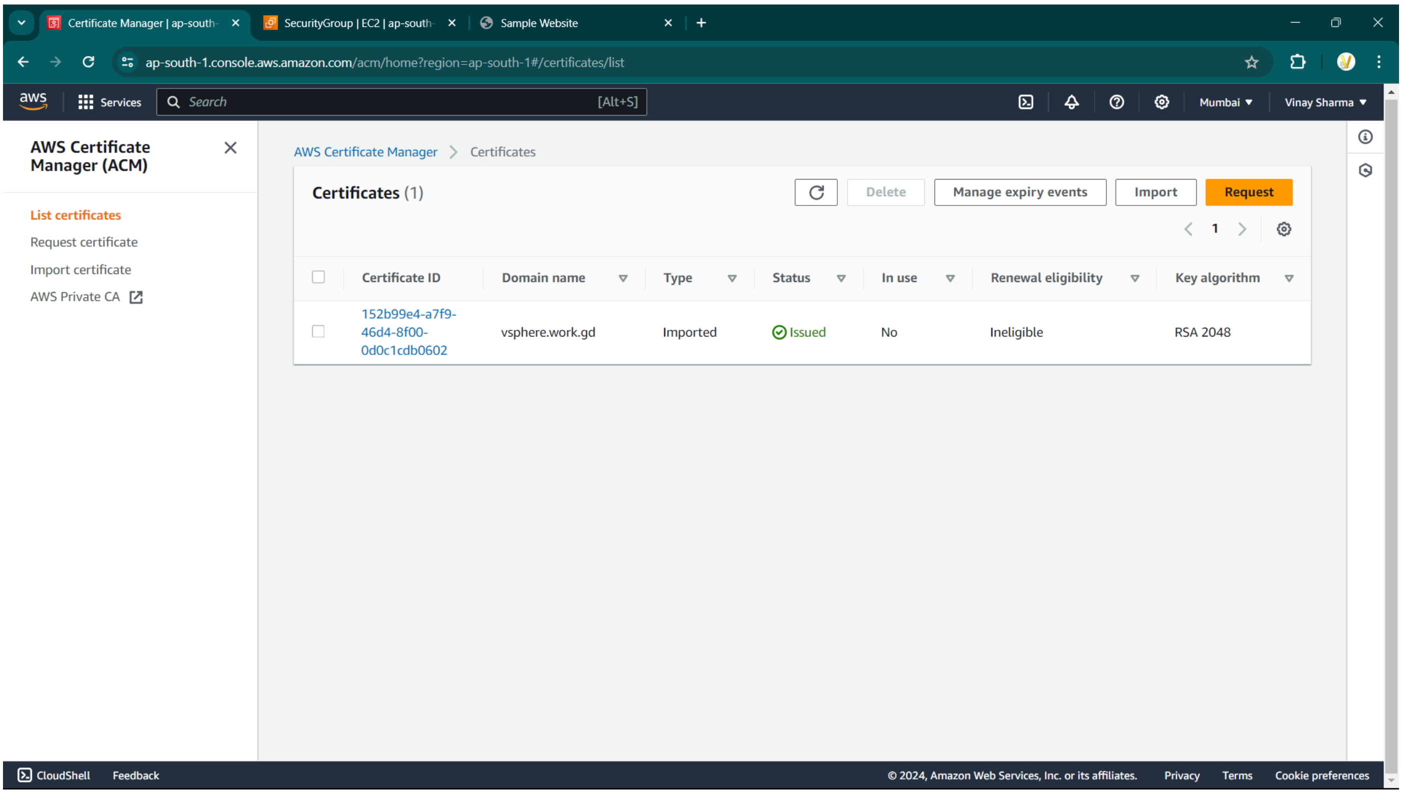
Certificate chain - *optional* Info

Paste the PEM-encoded certificate chain below.

- *Currently my website is got accessed using http protocol, which is probably not secure.*

- *I have generated SSL certificate for my website from domain registrar, so that my website runs on HTTPS.*

- *I have provided .req file to my domain registrar and then i got certificate keys from it; that i have put in Amazon Certificate Manager.*


Certificate Manager | ap-south-
SecurityGroup | EC2 | ap-south-
Sample Website
ap-south-1.console.aws.amazon.com/acm/home?region=ap-south-1#/certificates/list
Services
Search
[Alt+S]
Mumbai
Vinay Sharma

AWS Certificate Manager (ACM)

- List certificates
- Request certificate
- Import certificate
- AWS Private CA

AWS Certificate Manager > Certificates

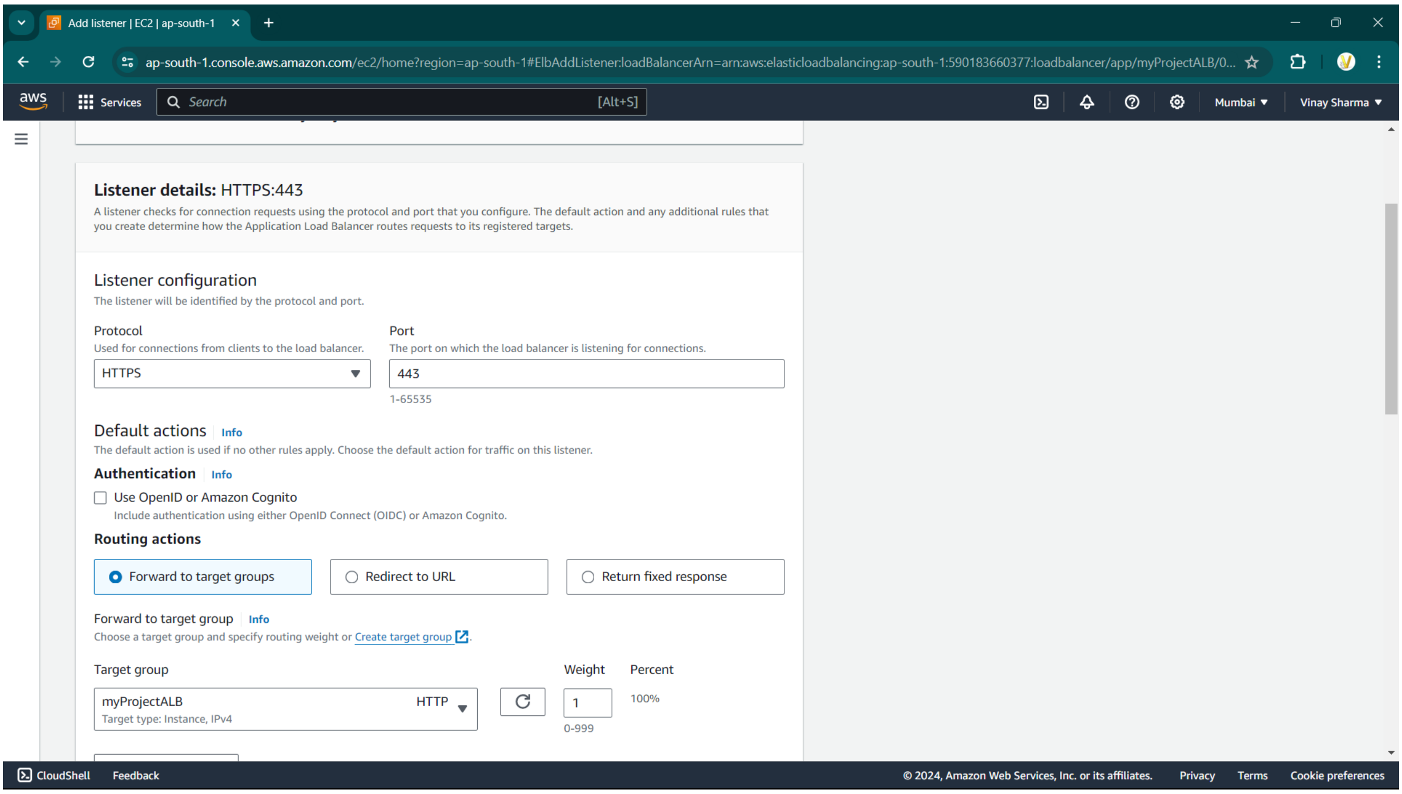
Certificates (1)

Delete
Manage expiry events
Import
Request
1

Certificate ID	Domain name	Type	Status	In use	Renewal eligibility	Key algorithm
152b99e4-a7f9-46d4-8f00-0d0c1cdb0602	vsphere.work.gd	Imported	✓ Issued	No	Ineligible	RSA 2048

CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

- *Here you can see, I have successfully imported my SSL certificate in ACM.*


Add listener | EC2 | ap-south-1
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ElbAddListener:loadBalancerArn=arn:aws:elasticloadbalancing:ap-south-1:590183660377:loadbalancer/app/myProjectALB/0...
Services
Search
[Alt+S]
Mumbai
Vinay Sharma

Listener details: HTTPS:443

A listener checks for connection requests using the protocol and port that you configure. The default action and any additional rules that you create determine how the Application Load Balancer routes requests to its registered targets.

Listener configuration

The listener will be identified by the protocol and port.

Protocol
Used for connections from clients to the load balancer.
HTTPS

Port
The port on which the load balancer is listening for connections.
443
1-65535

Default actions Info

The default action is used if no other rules apply. Choose the default action for traffic on this listener.

Authentication Info

□ Use OpenID or Amazon Cognito
Include authentication using either OpenID Connect (OIDC) or Amazon Cognito.

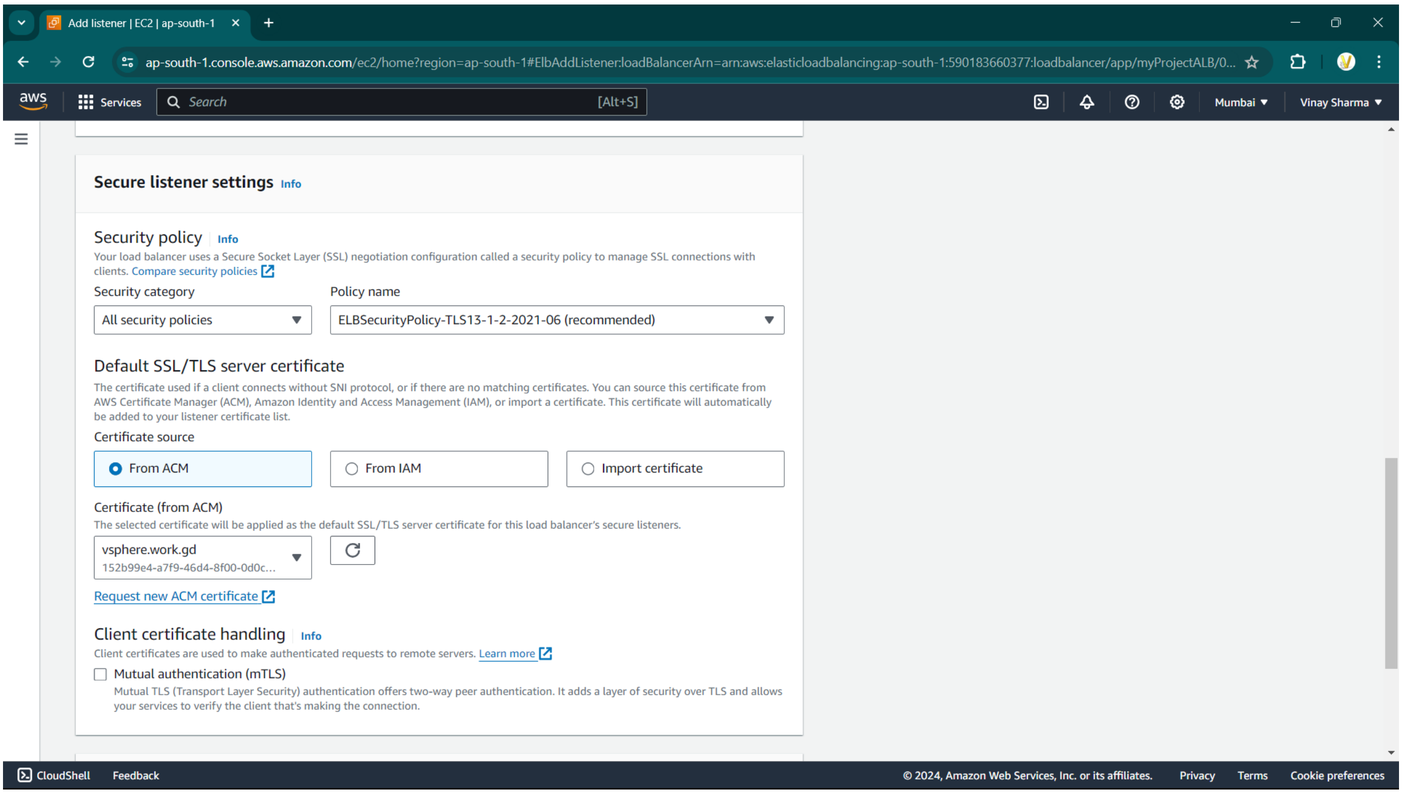
Routing actions

● Forward to target groups ○ Redirect to URL ○ Return fixed response

Forward to target group Info

Choose a target group and specify routing weight or Create target group.

Target group	Weight	Percent
myProjectALB HTTP Target type: Instance, IPv4	1 0-999	100%



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ElbAddListener:loadBalancerArn=arn:aws:elasticloadbalancing:ap-south-1:590183660377:loadbalancer/app/myProjectALB/0...

Secure listener settings Info

Security policy Info

Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration called a security policy to manage SSL connections with clients. Compare security policies

Security category: All security policies

Policy name: ELBSecurityPolicy-TLS13-1-2-2021-06 (recommended)

Default SSL/TLS server certificate

The certificate used if a client connects without SNI protocol, or if there are no matching certificates. You can source this certificate from AWS Certificate Manager (ACM), Amazon Identity and Access Management (IAM), or import a certificate. This certificate will automatically be added to your listener certificate list.

Certificate source

● From ACM ○ From IAM ○ Import certificate

Certificate (from ACM)

The selected certificate will be applied as the default SSL/TLS server certificate for this load balancer's secure listeners.

vsphere.work.gd
152b99e4-a7f9-46d4-8f00-0d0c...

Request new ACM certificate

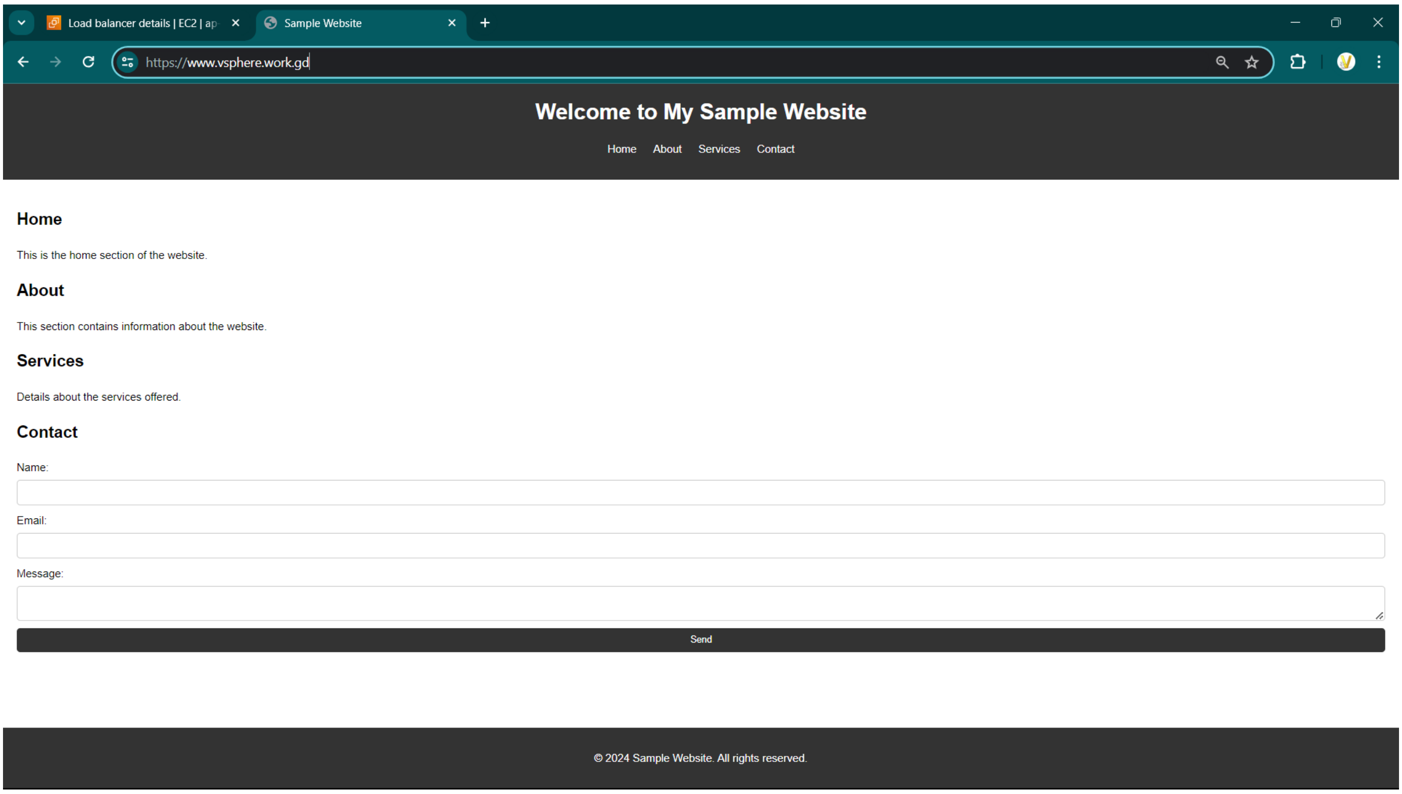
Client certificate handling Info

Client certificates are used to make authenticated requests to remote servers. Learn more

□ Mutual authentication (mTLS)

Mutual TLS (Transport Layer Security) authentication offers two-way peer authentication. It adds a layer of security over TLS and allows your services to verify the client that's making the connection.

- *Now in load balancer's listeners rules, remove rule for HTTP and add new rule that works on HTTPS and in Default SSL/TLS certificate, use ACM as a certificate source and select our imported certificate and click on save.*



https://www.vsphere.work.gd

Welcome to My Sample Website

Home

This is the home section of the website.

About

This section contains information about the website.

Services

Details about the services offered.

Contact

Name:

Email:

Message:

© 2024 Sample Website. All rights reserved.

- *Now you can see that our website is got accessed through HTTPS protocol.*

Implementation

The implementation of the project involved several key steps to deploy a highly available and secure web application on AWS. Below is a detailed account of the implementation process:

1. Setting Up the AWS Environment

- **AWS Account**: Ensure you have an AWS account set up and ready to use.
- **IAM Roles**: Create necessary IAM roles and policies to manage access and permissions.

2. Creating a Custom VPC

- **VPC Creation**: Navigate to the VPC Dashboard and create a custom VPC with a suitable IP address range (e.g., 10.0.0.0/16).
- **Subnets**: Create subnets within the VPC:
  - **Public Subnet**: For resources that need internet access (e.g., NAT Gateway).
  - **Private Subnets**: For hosting the web servers to enhance security.

3. Configuring Security Groups and NACLs

- **Security Groups**: Define security groups to control inbound and outbound traffic for your instances.

  - Web Server Security Group: Allow traffic from the load balancer only.
  - Load Balancer Security Group: Allow HTTP/HTTPS traffic from the internet.
- NACLs: Configure Network Access Control Lists (NACLs) to add an additional layer of security at the subnet level.

4. Setting Up the Load Balancer

- ELB Creation: Create an Elastic Load Balancer (ELB) through the EC2 Dashboard.
- Listeners and Target Groups: Configure listeners for HTTP and HTTPS. Create target groups and register your web servers with these target groups.
- SSL Certificate: Obtain an SSL certificate from AWS Certificate Manager (ACM) and attach it to the load balancer for HTTPS traffic.

5. Deploying Web Servers

- Launch EC2 Instances: Launch EC2 instances in the private subnets to host your web application.
- Web Server Configuration: Install necessary software (e.g., Apache, Nginx) and deploy your web application on these instances.
- Auto Scaling: Set up an Auto Scaling Group to automatically adjust the number of instances based on demand.

6. Configuring Route Tables and Internet Gateway

- Route Tables: Create and associate route tables for each subnet. Ensure the private subnets have routes to the NAT Gateway for internet access.

- Internet Gateway: Attach an internet gateway to your VPC and configure the public subnet route table to allow internet access.

7. DNS and Domain Configuration

- Domain Purchase: Purchase a domain through AWS Route 53 or another domain registrar.
- DNS Setup: Create a hosted zone in Route 53 and configure DNS records to point to the load balancer.

8. Testing and Validation

- Accessing the Application: Test accessing the web application through the load balancer’s public DNS name and ensure it routes traffic correctly.
- Security Testing: Perform security tests to ensure that the instances in private subnets are not directly accessible from the internet.
- Load Testing: Conduct load testing to verify that the application can handle expected traffic volumes.

By following these steps, the web application was successfully deployed on AWS with a robust architecture that ensures high availability, security, and scalability. The implementation process involved configuring various AWS services and components to create a secure and efficient environment for the application.

Conclusion

This project demonstrates the successful deployment of a highly available and secure web application on AWS. By leveraging an Elastic Load Balancer, multi-AZ deployment, and a custom VPC with private subnets, we ensured fault tolerance, scalability, and network isolation. Implementing security groups, NACLs, a custom domain, and an SSL certificate further enhanced the security and reliability of the application. Overall, this architecture provides a robust and secure environment, ensuring a seamless and secure experience for users.

References

- Youtube
- Udemy
- AWS Solutions Architect Study Guide v4