

SECURE WEBSITE HOSTING WITH APPLICATION LOAD BALANCER (ALB), SSL CERTIFICATE AND AUTO SCALING ON AWS

SUMMERY :

The Project Involves Hosting A Secure And Scalable Website On Aws Using EC2 Instances, Application Load Balancer (ALB), SSL Certificates, And Auto Scaling. The Website Will Be Hosted On EC2 Instances, Which Will Be Automatically Scaled Based On Traffic Demands. Application Load Balancer Will Manage And Distribute Incoming Traffic To Ensure High Availability And Load Balancing. SSL Certificates, Managed Through AWS Certificate Manager (ACM), Will Secure The Website With HTTPS. Route 53 Will Handle DNS Management, Directing Traffic To The Application Load Balancer. The Outcome Is A Highly Available, Secure, And Resilient Website Capable Of Efficiently Managing Varying Traffic Loads. A Secure, Resilient, And Scalable Website Hosted On AWS, Capable Of Handling Varying Traffic Loads Efficiently And Ensuring High Availability.

- ❖ Log Into Aws Console
- ❖ Navigate To EC2 Service → Instances → Launch Templates → Click On Create Launch Template Button

The screenshot shows the AWS EC2 Launch Templates page. The URL is ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchTemplates. The left sidebar has a tree view with 'Instances' expanded, showing 'Launch Templates' selected. The main content area has a heading 'EC2 launch templates' and sub-headings 'Streamline, simplify and standardize instance launches'. It includes a section 'Benefits and features' with two items: 'Streamline provisioning' and 'Simplify permissions'. A large orange button labeled 'Create launch template' is prominent. The bottom right corner shows a footer with links to 'Documentation', 'API reference', and social media icons.

- ❖ Launch Template Configurations :
- ❖ Template Name : **Home_Template**

The screenshot shows the 'Create launch template' wizard. The URL is ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate. The left sidebar shows 'EC2 > Launch templates > Create launch template'. The main form has a section 'Launch template name and description' where 'Home_Template' is entered. Below it is a 'Template version description' field with 'Home_Template' and a note about auto-scaling. A summary panel on the right lists 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', and 'Storage (volumes)'. A tooltip for the 'Free tier' is open, stating: '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'. At the bottom are 'Cancel' and 'Create launch template' buttons.

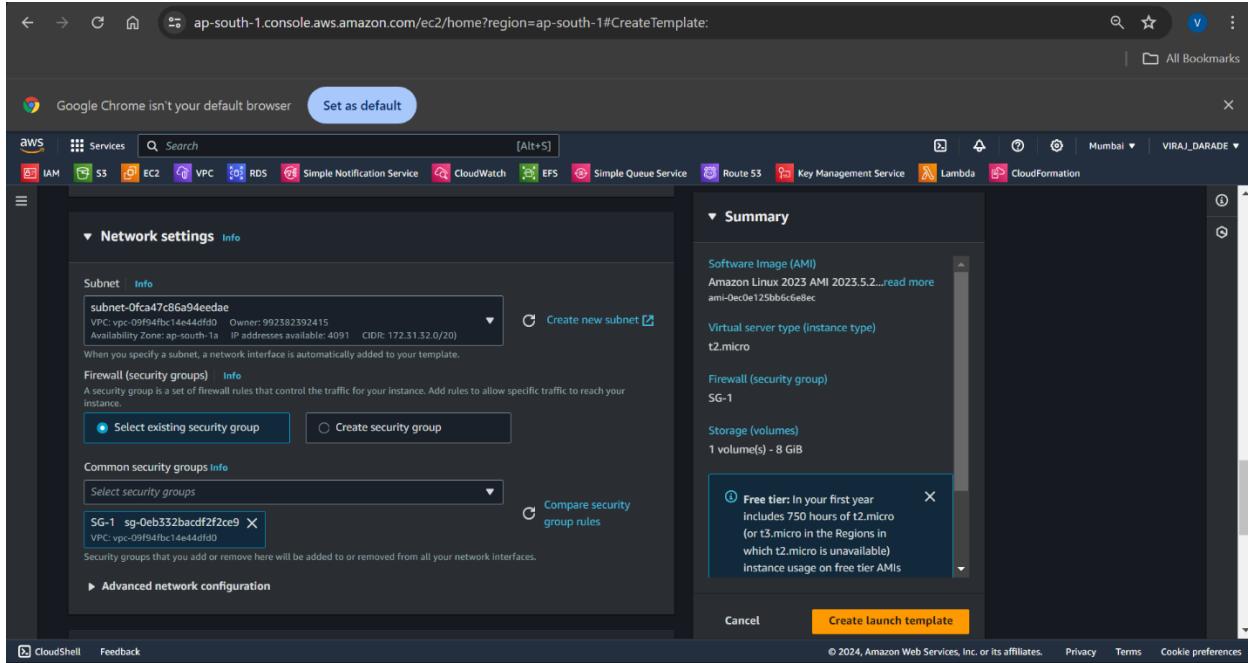
❖ Amazon Machine Image(Ami) : Amazon Linux

The screenshot shows the AWS Management Console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTemplate. The top navigation bar includes services like IAM, S3, EC2, VPC, RDS, Simple Notification Service, CloudWatch, EFS, Simple Queue Service, Route 53, Key Management Service, Lambda, and CloudFormation. The main content area is titled "Application and OS Images (Amazon Machine Image)". It displays a search bar and a "Quick Start" section with icons for Amazon Linux, macOS, Ubuntu, Windows, and Red Hat. Below this is a section for "Amazon Machine Image (AMI)" specifically for "Amazon Linux 2023 AMI". A tooltip for the "Free tier" is visible, stating: "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". At the bottom right of the main panel is a "Create launch template" button.

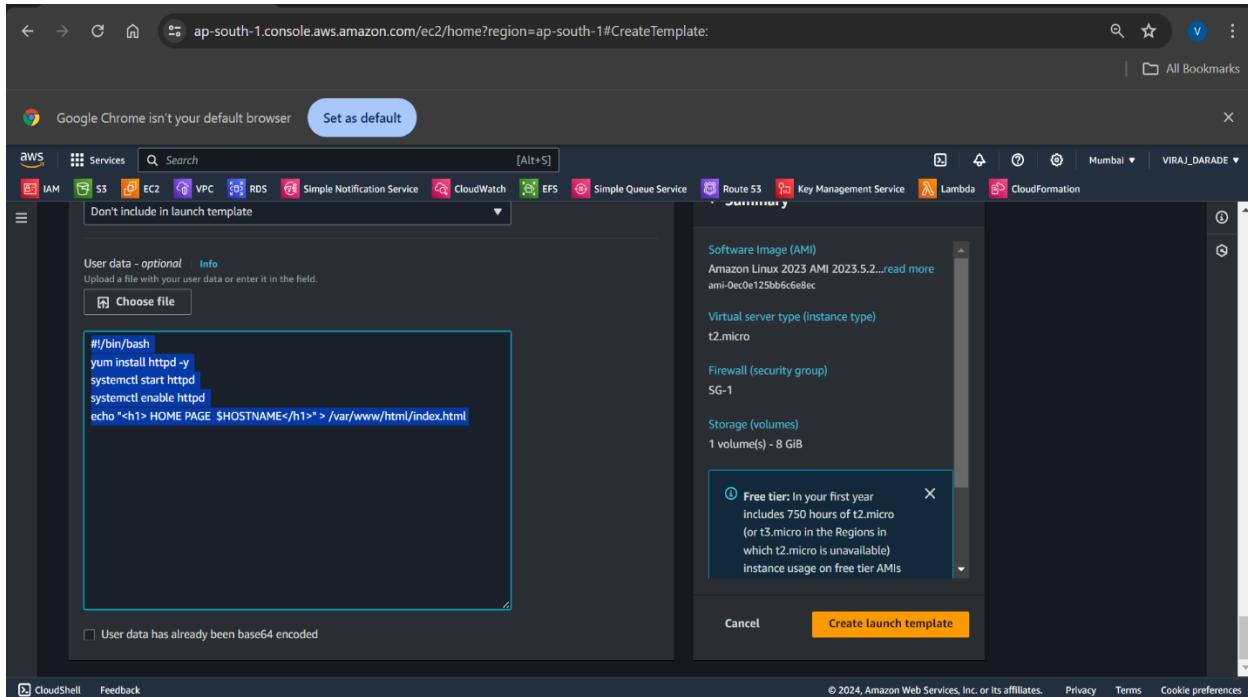
- ❖ Instance Type : **t2.micro**
- ❖ Key Pair : **Mumbai-Key**

The screenshot shows the same AWS Management Console page for creating a launch template. The "Instance type" section is now selected, showing "t2.micro" as the chosen option. The "Key pair (login)" section shows "mumbai-key" selected as the key pair. A tooltip for the "Free tier" is visible, stating: "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". The "Create launch template" button is again at the bottom right.

❖ Security Group : SG-1 (Already Created)



- ❖ Add User Data For Home Webpage
- ❖ Review & Click On Create Launch Template Button



- ❖ Repeat Same Process For **Laptop & Mobile Templates**
- ❖ Only Change **Template Name & User Data**
- ❖ Here We Can See, Our **Templates Will Be Created**
- ❖ **Templates : Home_Template, Laptop_Template & Mobile_Template**

The screenshot shows the AWS EC2 Launch Templates page. The left sidebar navigation includes EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates (selected), Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area displays a table titled "Launch Templates (1/3)" with the following data:

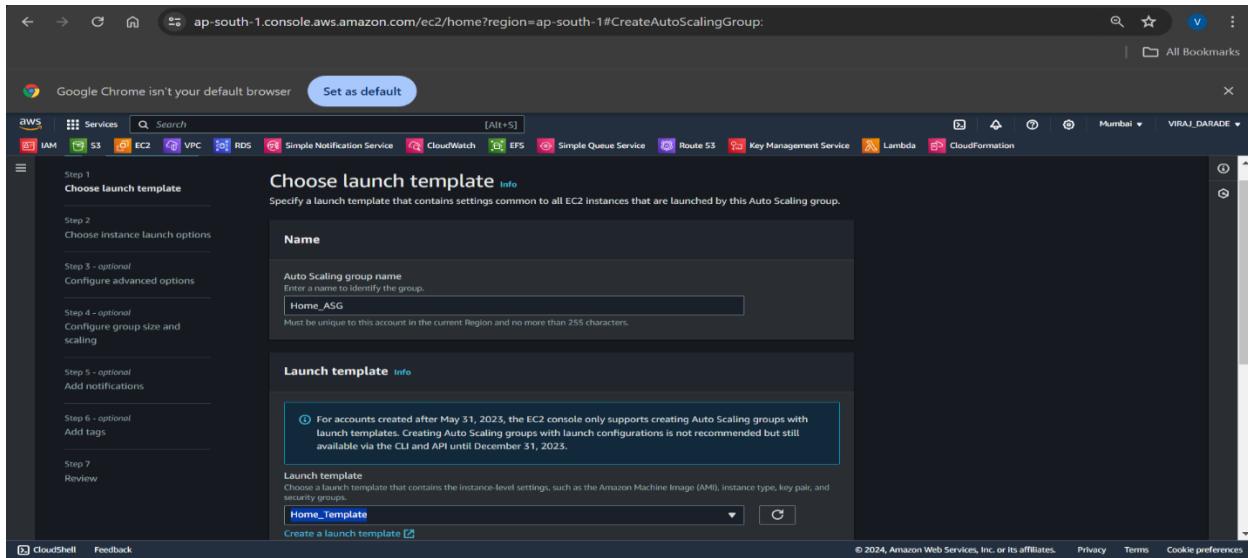
Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By
lt-0df1c70c7d2546926	Home_Template	1	1	2024-07-12T06:48:39.000Z	arn:aws:iam::99238239...
lt-0a0d70df03805de0f	Laptop_Template	1	1	2024-07-12T06:51:37.000Z	arn:aws:iam::99238239...
lt-0ce405af13eccfa46	Mobile_Template	1	1	2024-07-12T06:54:00.000Z	arn:aws:iam::99238239...

A modal window titled "Home_Template (lt-0df1c70c7d2546926)" is open, showing "Launch template details" with the launch template ID "lt-0df1c70c7d2546926", name "Home_Template", default version "1", owner "arn:aws:iam::992382392415:root", and tabs for Details, Versions, and Template tags.

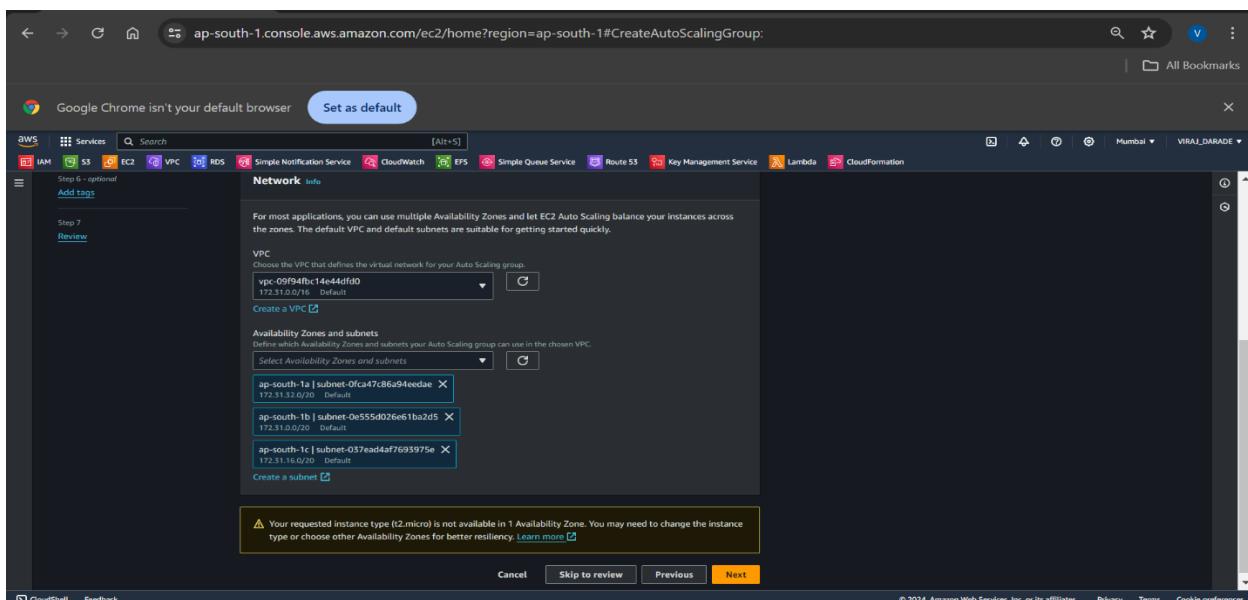
- ❖ Navigate To **Auto Scaling** → Auto Scaling Groups → Click On **Create Auto Scaling Group** Button

The screenshot shows the AWS EC2 Auto Scaling Groups page. The left sidebar navigation includes Snapshots, Lifecycle Manager, Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), Load Balancing (Load Balancers, Target Groups, Trust Stores), Auto Scaling (Auto Scaling Groups selected), and Settings. The main content area features a large banner with the text "Amazon EC2 Auto Scaling helps maintain the availability of your applications". Below the banner, there are sections for "How it works" (illustrated with a diagram of a cloud icon labeled "Auto Scaling group") and "Pricing" (text stating "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."). A prominent orange "Create Auto Scaling group" button is located in the center-right of the main content area.

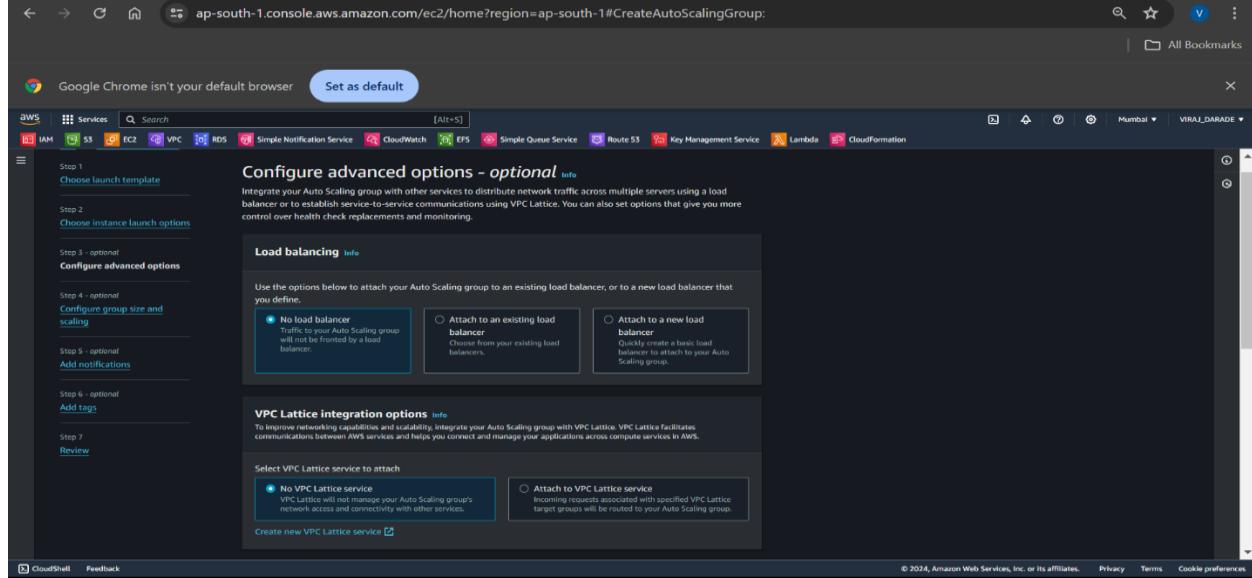
- ❖ Auto Scaling Configurations :
- ❖ Auto Scaling Group Name : **Home_ASG**
- ❖ Launch Template : Select Created **Home_Template**
- ❖ Click On **Next** Button



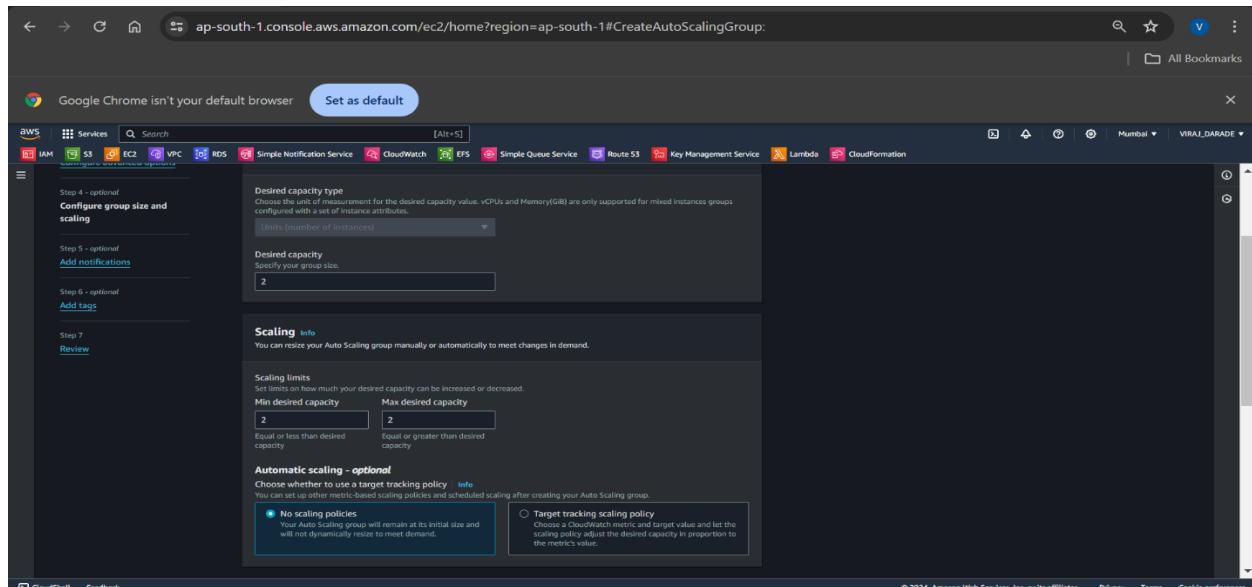
- ❖ VPC : Select **Default VPC**
- ❖ Availability Zones & Subnets : Select **All Subnets**
- ❖ Click On **Next** Button



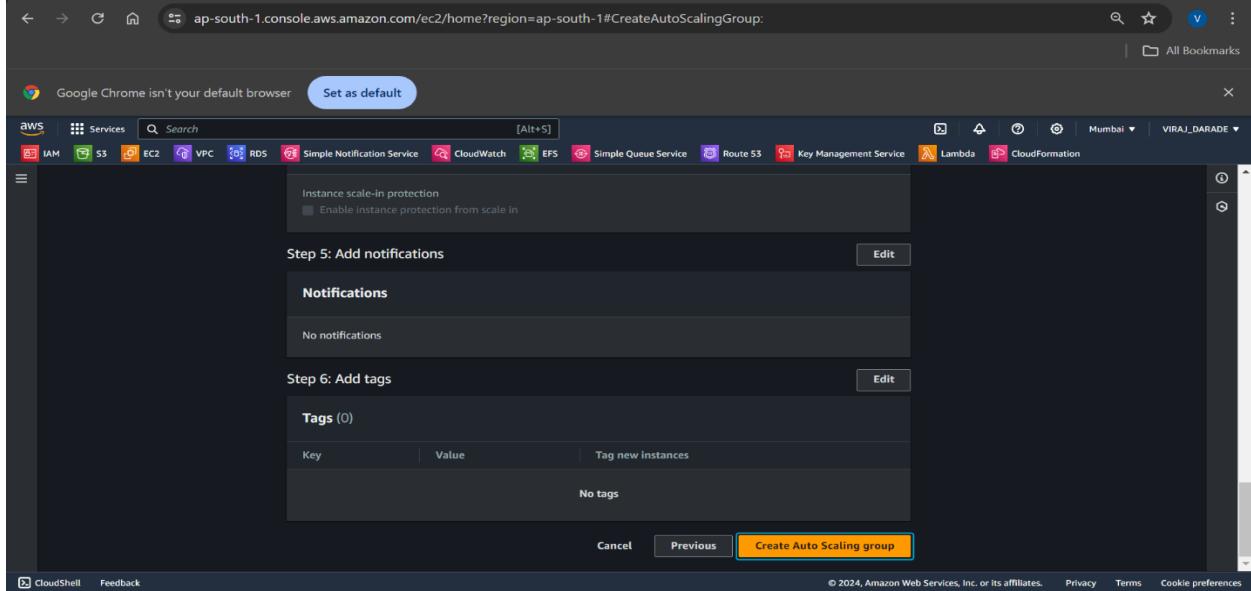
- ❖ Load Balancing : Select **No Load Balancer** (We Attach Load Balancer Later)
- ❖ VPC Lattice Integration Options : Select **No VPC Lattice Service**
- ❖ Click On **Next Button**



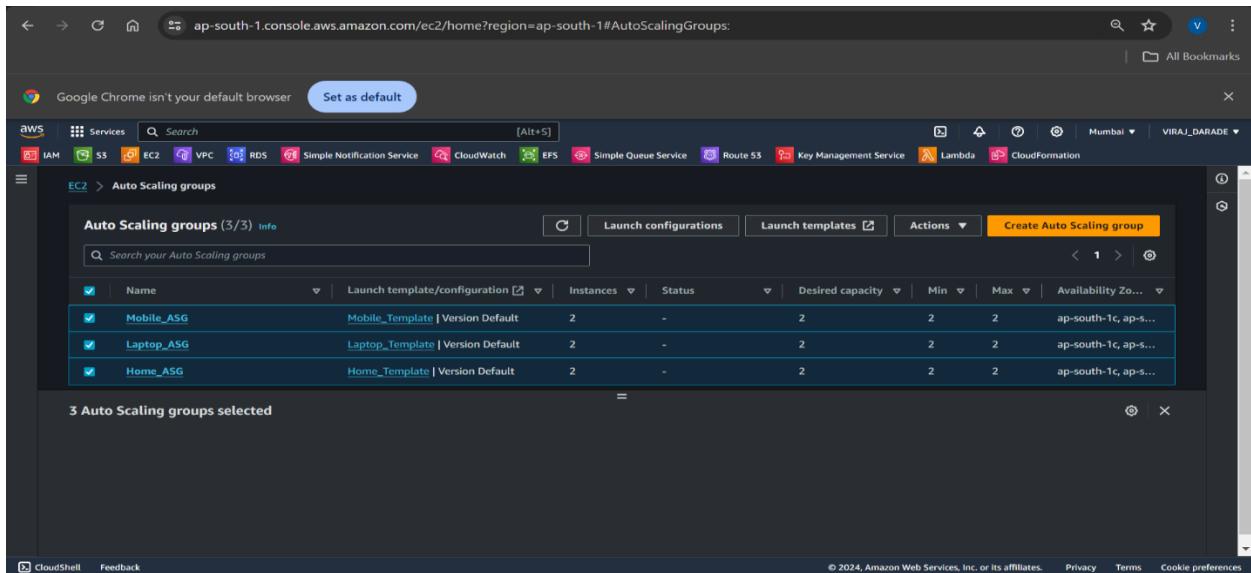
- ❖ Desired Capacity : **2**
- ❖ Min Desired Capacity : **2**
- ❖ Max Desired Capacity : **2**
- ❖ Automatic Scaling : Select **No Scaling Policies** (Because Static Mode Of Instances)



- ❖ Add Notifications: **Optional**
- ❖ Add Tags : **Optional**
- ❖ Click On Create Auto Scaling Group Button



- ❖ Repeat Same Process For **Laptop_ASG & Mobile_ASG**
- ❖ Only Change Auto Scaling Group Name & Launch Template
- ❖ Here We Can See, Our Auto Scaling Groups Will Be Created
- ❖ Auto Scaling Groups : **Home_ASG, Laptop_ASG & Mobile_ASG**



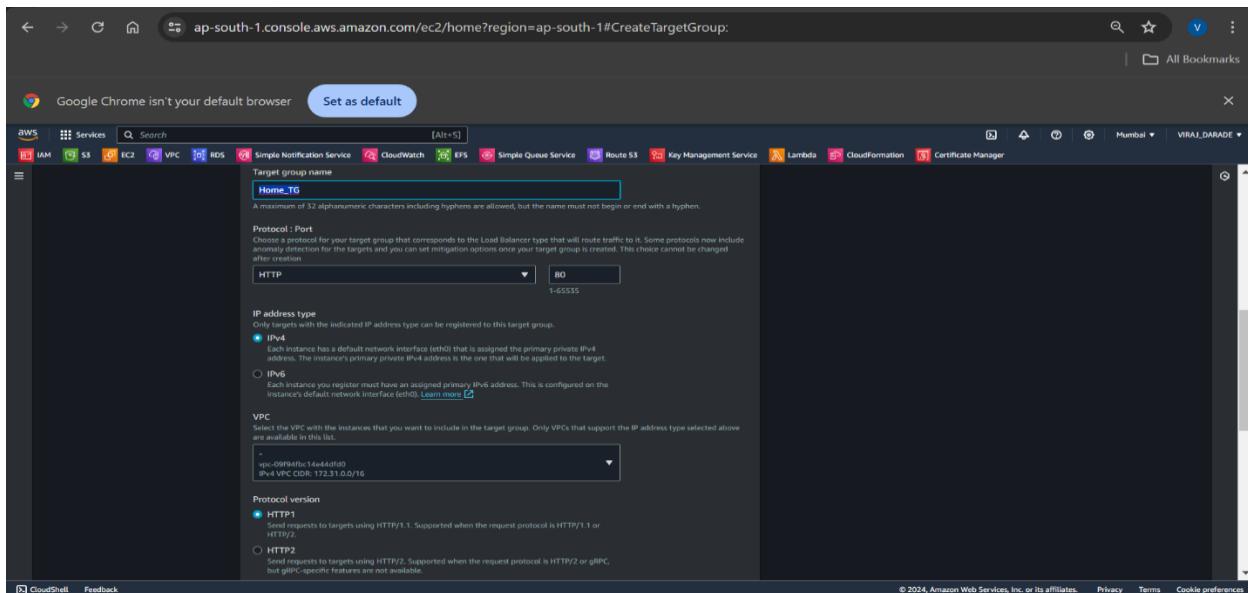
❖ Navigate To Load Balancing → Target Groups → Click On Create Target Group Button

The screenshot shows the AWS Management Console interface for the EC2 service, specifically the Target Groups section. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#TargetGroups:`. The left sidebar has 'Target Groups' selected under the 'Load Balancing' category. The main content area displays a table titled 'Target groups' with one row: 'No target groups'. Below the table, a message says 'You don't have any target groups in ap-south-1'. At the top right of the table, there is a yellow button labeled 'Create target group'.

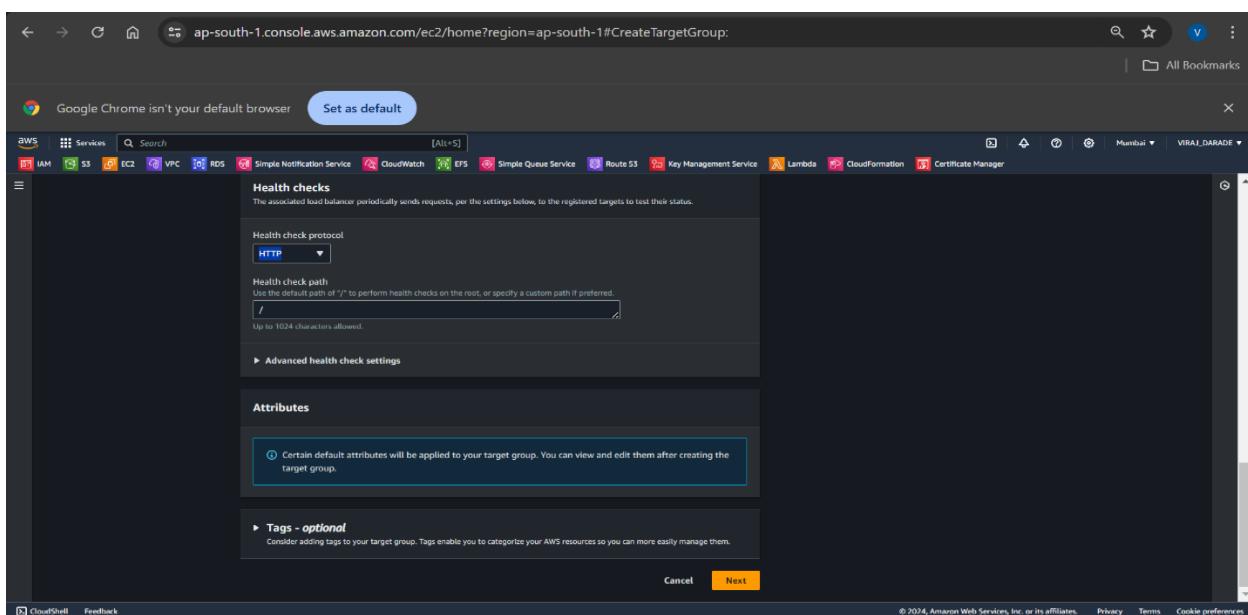
❖ Target Group Configurations :
❖ Target Type : Select Instances

The screenshot shows the 'Specify group details' step of the 'Create Target Group' wizard. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTargetGroup:`. The left sidebar shows 'Step 1: Specify group details' and 'Step 2: Register targets'. The main content area is titled 'Specify group details' and contains a section titled 'Basic configuration'. It says 'Your load balancer routes requests to the targets in a target group and performs health checks on the targets.' Below this, there is a heading 'Choose a target type' with four options: 'Instances' (selected), 'IP addresses', 'Lambda function', and 'Application Load Balancer'. Each option has a brief description of its features.

- ❖ Target Group Name : **Home_TG**
- ❖ Protocol : **HTTP**
- ❖ Port : **80**
- ❖ IP Address Type : **IPv4**
- ❖ VPC : Select **Default VPC**
- ❖ Protocol Version : **HTTP1**



- ❖ Health Check Protocol : **HTTP**
- ❖ Health Check Path : **/**



❖ Register Targets : Don't Select Instances (Because We Apply Auto Scaling)

The screenshot shows the 'Register targets' step in the AWS EC2 Target Groups creation wizard. The top navigation bar includes links for IAM, S3, EC2, VPC, RDS, Simple Notification Service, CloudWatch, EFS, Simple Queue Service, Route 53, Key Management Service, Lambda, CloudFormation, and Certificate Manager. The main content area is titled 'Register targets' and contains the following information:

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 (5)

Instance ID	Name	State	Security groups	Zone	Private IPv4 address
i-03acd9303f134d879		Running	SG-1	ap-south-1a	172.31.33.44
i-099105fb08fe01377		Running	SG-1	ap-south-1b	172.31.4.223
i-0c7e30d709443248c		Running	SG-1	ap-south-1a	172.31.42.190
i-007b845702d0f3b4d		Running	SG-1	ap-south-1b	172.31.0.32
i-0fde8becab4782b02		Running	SG-1	ap-south-1a	172.31.37.255

Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with commas)
Include as pending below

❖ Click On Create Target Group Button

The screenshot shows the 'Review targets' step in the AWS EC2 Target Groups creation wizard. The top navigation bar is identical to the previous screenshot. The main content area is titled 'Review targets' and contains the following information:

Targets (0)

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
No instances added yet Specify instances above, or leave the group empty if you prefer to add targets later.								

0 pending

Create target group

- ❖ Repeat Same Process For Laptop-TG & Mobile-TG
- ❖ Only Change Target Group Name & Health Check Path
- ❖ Here We Can See, Our Target Groups Will Be Created

The screenshot shows the AWS EC2 Target Groups page. The left sidebar includes options like Instances, Images, and Auto Scaling. The main content area displays a table titled "Target groups (3/3) info". The table has columns for Name, ARN, Port, Protocol, Target type, Load balancer, and VPC ID. Three target groups are listed:

Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
Mobile-TG	arn:aws:elasticloadbalancing:ap-south-1:912345678901:targetgroup/Mobile-TG/54321	80	HTTP	Instance	None associated	vpc-09f94fb14e44dfd0
Laptop-TG	arn:aws:elasticloadbalancing:ap-south-1:912345678901:targetgroup/Laptop-TG/54321	80	HTTP	Instance	None associated	vpc-09f94fb14e44dfd0
Home-TG	arn:aws:elasticloadbalancing:ap-south-1:912345678901:targetgroup/Home-TG/54321	80	HTTP	Instance	None associated	vpc-09f94fb14e44dfd0

At the bottom of the table, it says "3 target groups selected".

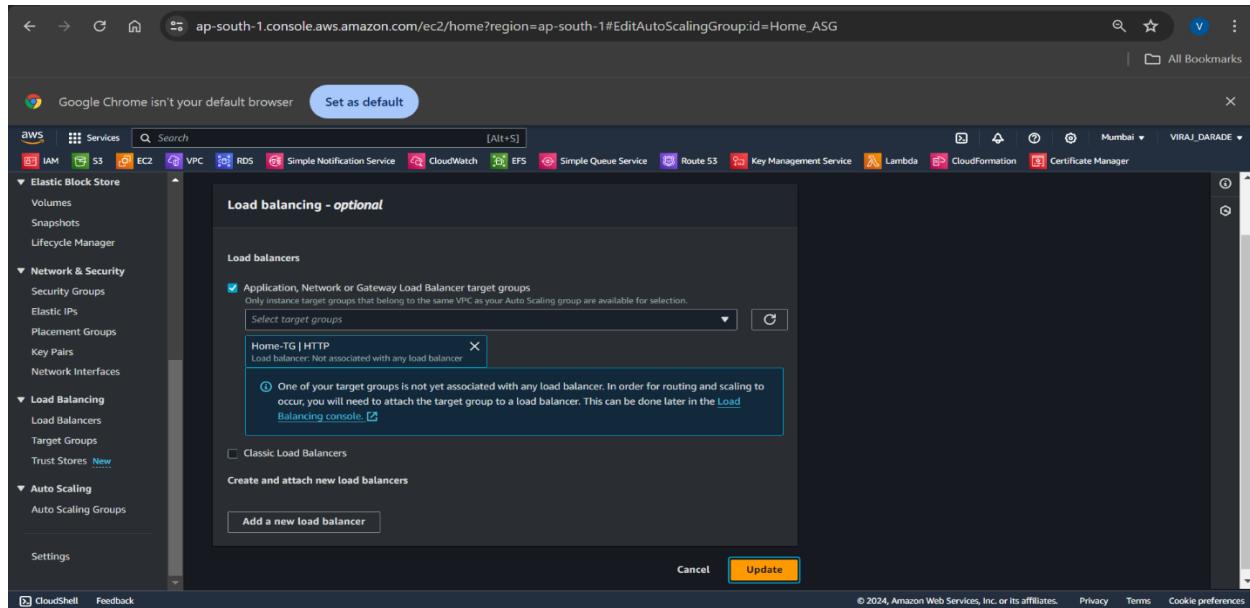
- ❖ Navigate To Auto Scaling Group To Apply Load Balancing
- ❖ Select Home_ASG, Scroll Down To Load Balancing And Click On Edit Button

The screenshot shows the AWS Auto Scaling Groups page. The left sidebar includes options like Snapshots, Lifecycle Manager, Network & Security, Load Balancing, and Auto Scaling. The main content area displays a table titled "Auto Scaling groups (1/3) info". The table has columns for Name, Launch template/configuration..., Instances, Status, Desired capacity, Min, Max, and Available. Three auto scaling groups are listed:

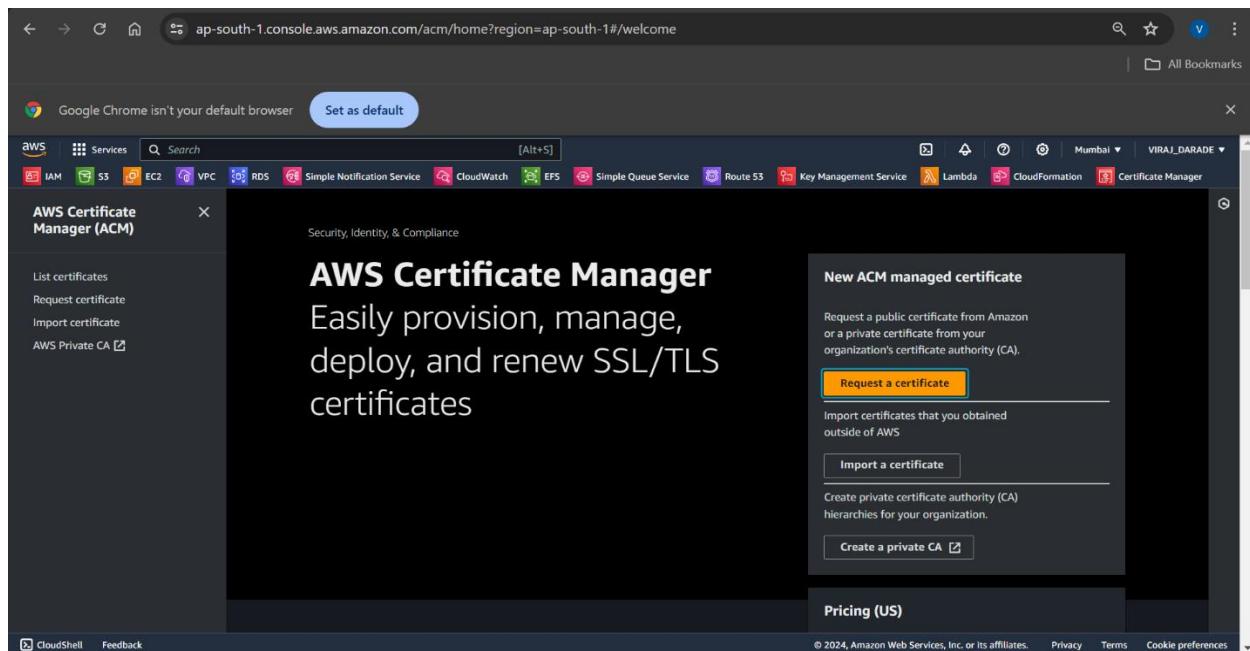
Name	Launch template/configuration...	Instances	Status	Desired capacity	Min	Max	Available...
Home_ASG	Home_Template Version Default	2	-	2	2	2	ap-south...
Laptop_ASG	Laptop_Template Version Default	2	-	2	2	2	ap-south...
Mobile_ASG	Mobile_Template Version Default	2	-	2	2	2	ap-south...

Below the table, under "Auto Scaling group: Home_ASG", there is a "Load balancing" section. It contains two tabs: "Load balancer target groups" and "Classic Load Balancers". An "Edit" button is located at the top right of this section.

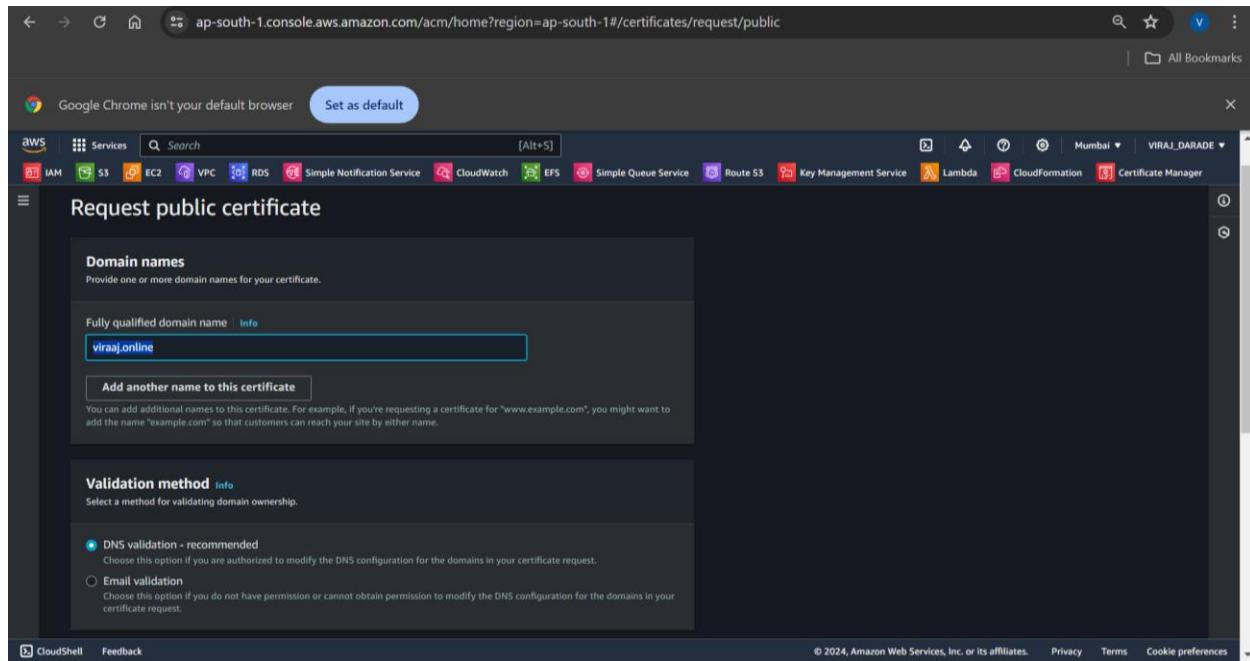
- ❖ Load Balancers : Check The Check Box Of Application Load Balancer
- ❖ Target Group : Select Home-TG
- ❖ Click On Update Button
- ❖ Repeat Same Process For Laptop_ASG & Mobile_ASG
- ❖ Only Change Target Group



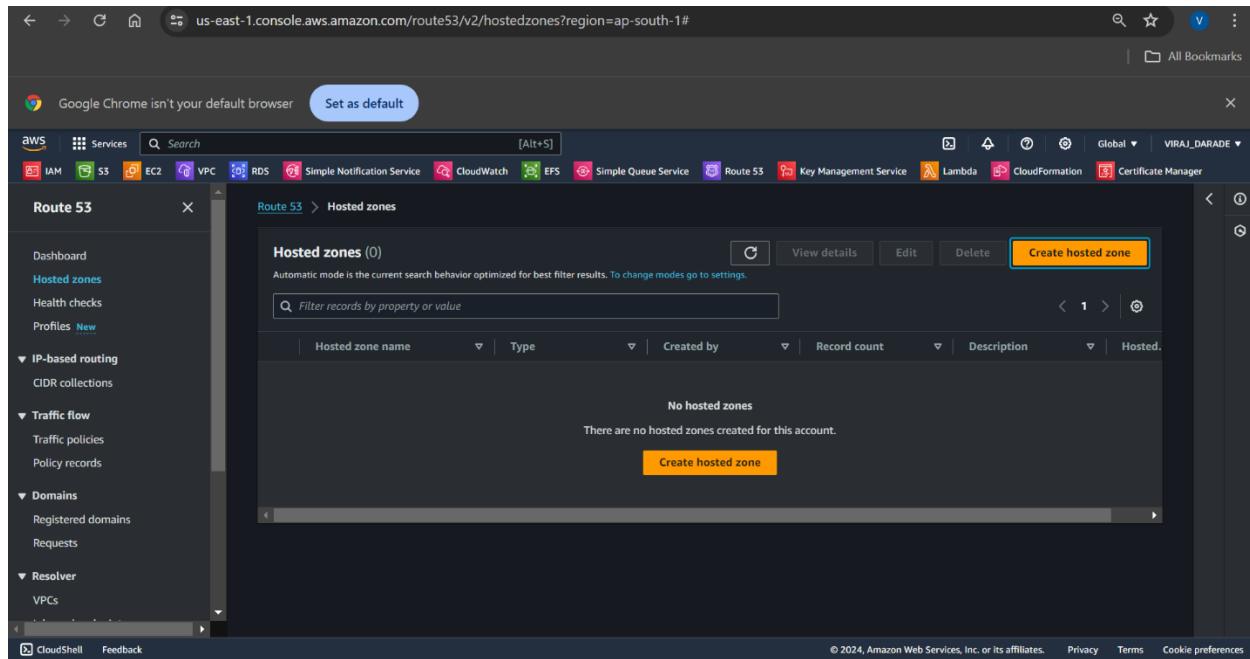
- ❖ Navigate To Aws Certificate Manager (ACM) Service
- ❖ Click On Request A Certificate Button



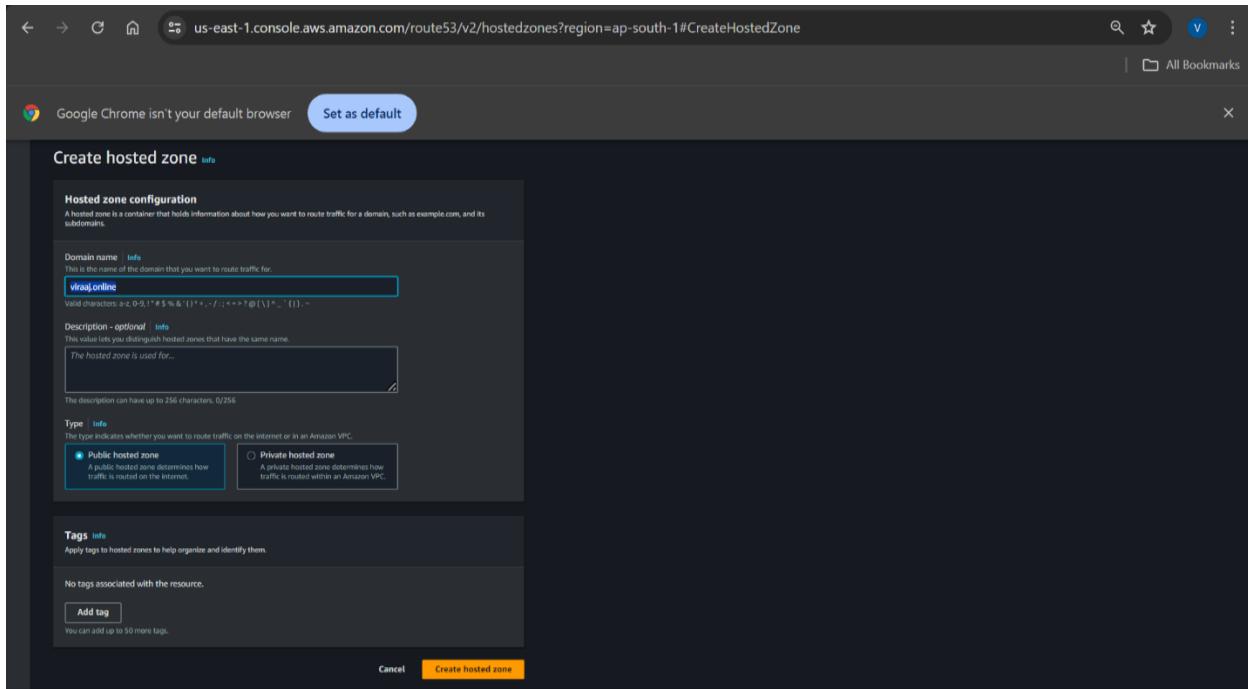
- ❖ Domain Name : **viraaj.online** (Purchased From **Hostinger**)
- ❖ Validation Method : Select **DNS Validation – Recommended**



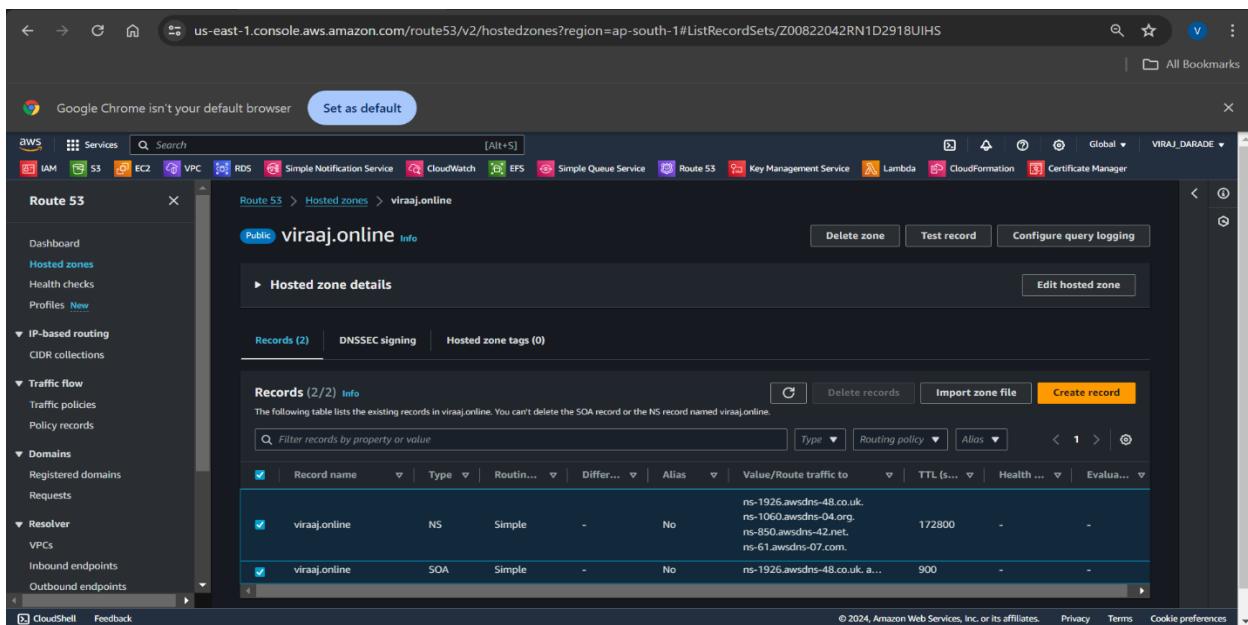
- ❖ Navigate To **Route 53 Service** → Click On **Hosted Zones** → Click On **Create Hosted Zone** Button



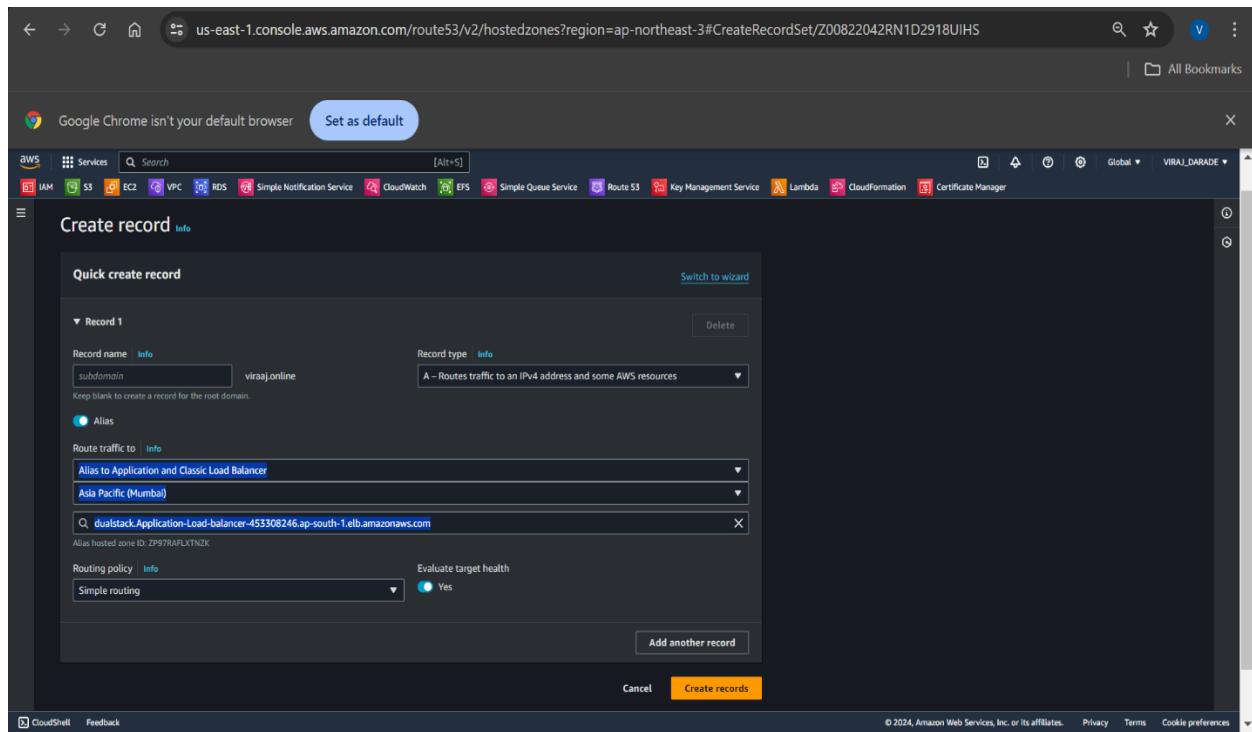
- ❖ Hosted Zone Configurations :
- ❖ Domain Name : **viraaj.online**
- ❖ Type : Select **Public Hosted Zone**
- ❖ Click On **Create Hosted Zone** Button



- ❖ Here We Can See, Our **Hosted Zone** Will Be Created
- ❖ Records : **NS & SOA**, By Default Created With **Hosted Zone**



- ❖ **Create 1 Record By Using Alias :**
- ❖ **Record Name : Leave Blank**
- ❖ **Record Type : Select A (For IPv4 Addresses)**
- ❖ **Turn Alias On**
- ❖ **Route Traffic To : Select Alias To Application & Classic Load Balancer**
- ❖ **Region : Asia Pacific (Mumbai)**
- ❖ **Routing Policy : Select Simple Routing**
- ❖ **Click On Create Record Button**



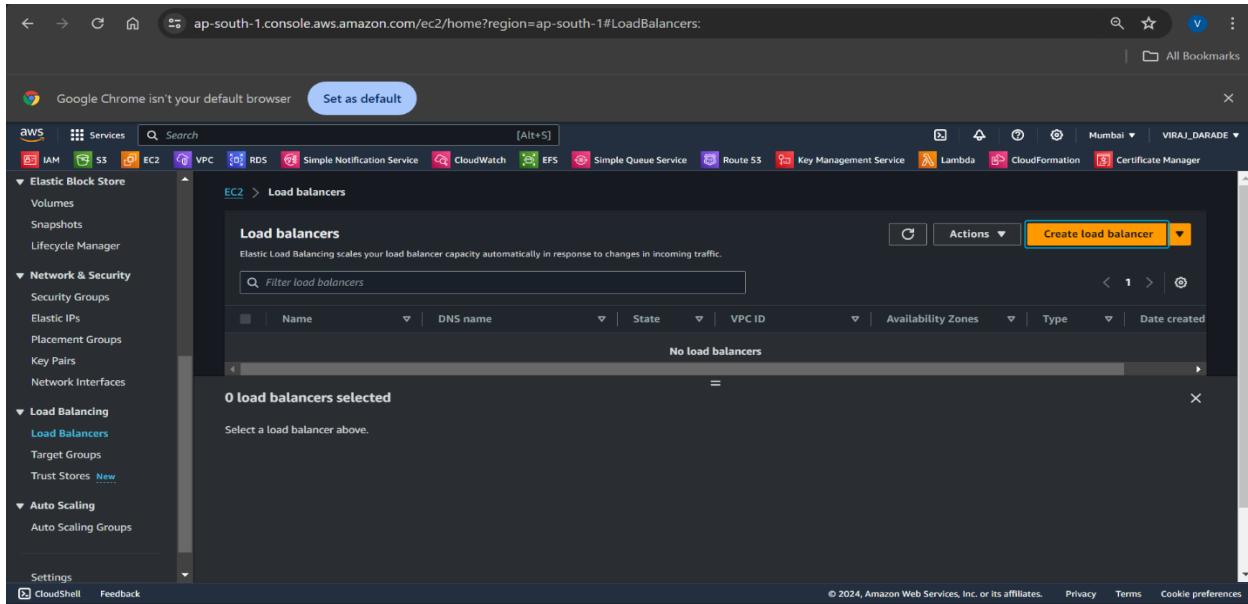
- ❖ Here We Can See Our Record Will Be Created & We Also Created The Record For SSL Certificate By Using Cname
- ❖ Cname & Value We Get From Aws Certificate Manager

Record name	Type	Routing policy	Alias	Value/Route traffic to	TTL (s.)
viraaj.online	A	Simple	-	dualstack.application-load-b...	-
viraaj.online	NS	Simple	-	ns-1926.awsdns-48.co.uk. ns-1060.awsdns-04.org. ns-850.awsdns-42.net. ns-61awsdns-07.com.	172800
viraaj.online	SOA	Simple	-	ns-1926.awsdns-48.co.uk. a...	900
_949209235b5daad1beb67df1df074865.vir...	CNAME	Simple	-	_b2c8db573cb51429796332...	50

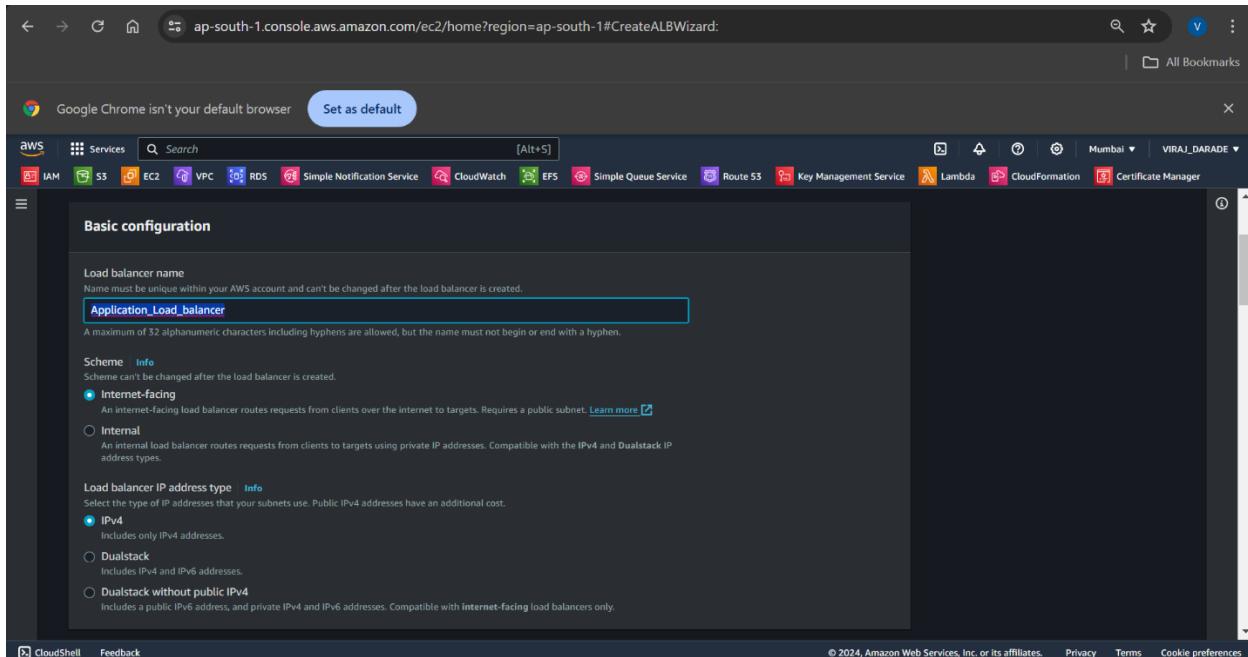
- ❖ Here We Can See, Our Certificate Will Be Issued

Certificate ID	Domain name	Type	Status	In use
17bc5567-ba46-47dc-8124-f24cd8744b80	viraaj.online	Amazon Issued	Issued	No

❖ Navigate To Load Balancing → Load Balancers → Click On Create Load Balancer



- ❖ Application Load Balancer Configurations :
- ❖ Load Balancer Name : **Application_Load_Balancer**
- ❖ Scheme : **Internet-Facing**
- ❖ Load Balancer IP Address Type : **IPv4**



- ❖ Network Mapping :
- ❖ VPC : Select Default VPC
- ❖ Mappings : Select All Subnets

The screenshot shows the 'Network mapping' step of the Create ALB Wizard. In the 'VPC' dropdown, 'ap-south-1' is selected. Under 'Mappings', three subnets are listed: 'ap-south-1a (ap-s1-a2)', 'ap-south-1b (ap-s1-a2)', and 'ap-south-1c (ap-s1-a2)'. Each subnet has its 'IPv4 address' and 'Assigned by AWS' status indicated.

- ❖ Security Groups: Select Default & SG-1 (Already Created)
- ❖ Listeners & Routing :
- ❖ Protocol : HTTPS
- ❖ Port : 443
- ❖ Forward To : Select Home-TG

The screenshot shows the 'Listeners and routing' step of the Create ALB Wizard. A listener for 'HTTPS:443' is configured to forward traffic to the 'Home-TG' target group. The 'Protocol' is set to 'HTTPS' and the 'Port' is '443'. The 'Default action' is 'Forward to Home-TG Target type: Instance, IPv4'.

- ❖ Secure Listener Settings:
- ❖ Default SSL/TLS Server Certificate :
- ❖ Certificate Source : Select From ACM
- ❖ Certificate (From ACM) : Select **viraaj.online** Certificate

The screenshot shows the AWS CloudFront Create Distribution Wizard - Step 2: Set up security settings. The 'Default SSL/TLS server certificate' section is highlighted, showing 'From ACM' selected and 'viraaj.online' chosen from the dropdown.

- ❖ Review & Click On **Create Load Balancer** Button

The screenshot shows the AWS CloudFront Create Distribution Wizard - Step 3: Review configuration. The 'Listeners and routing' section is highlighted, showing the configuration for port 443.

❖ Here We Can See, Our Load Balancer Will Be Created (Application Load Balancer)

The screenshot shows the AWS Management Console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LoadBalancers. The left sidebar is collapsed. The main area displays the 'Load balancers' section with one item listed:

Name	DNS name	Status	VPC ID	Availability Zones	Type	Date created
Application-Load-balancer...	Application-Load-balancer...	Active	vpc-09f94fb14e44df0	3 Availability Zones	application	July 12, 2024

Below the table, a modal window titled 'Load balancer: Application-Load-balancer' is open, showing the 'Details' tab with the following information:

Load balancer type	Status	VPC	Load balancer IP address type
Application	Active	vpc-09f94fb14e44df0	IPv4

At the bottom right of the modal, there are links for 'Privacy', 'Terms', and 'Cookie preferences'.

❖ Scroll Down To Listeners & Rules

❖ Select **HTTPS:443** → Manage Rules → Add Rule

The screenshot shows the same AWS Management Console URL as the previous screenshot. The left sidebar is collapsed. The main area displays the 'Listeners and rules' section for the 'HTTPS:443' listener:

Listeners and rules (1/1) Info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the defined rules.

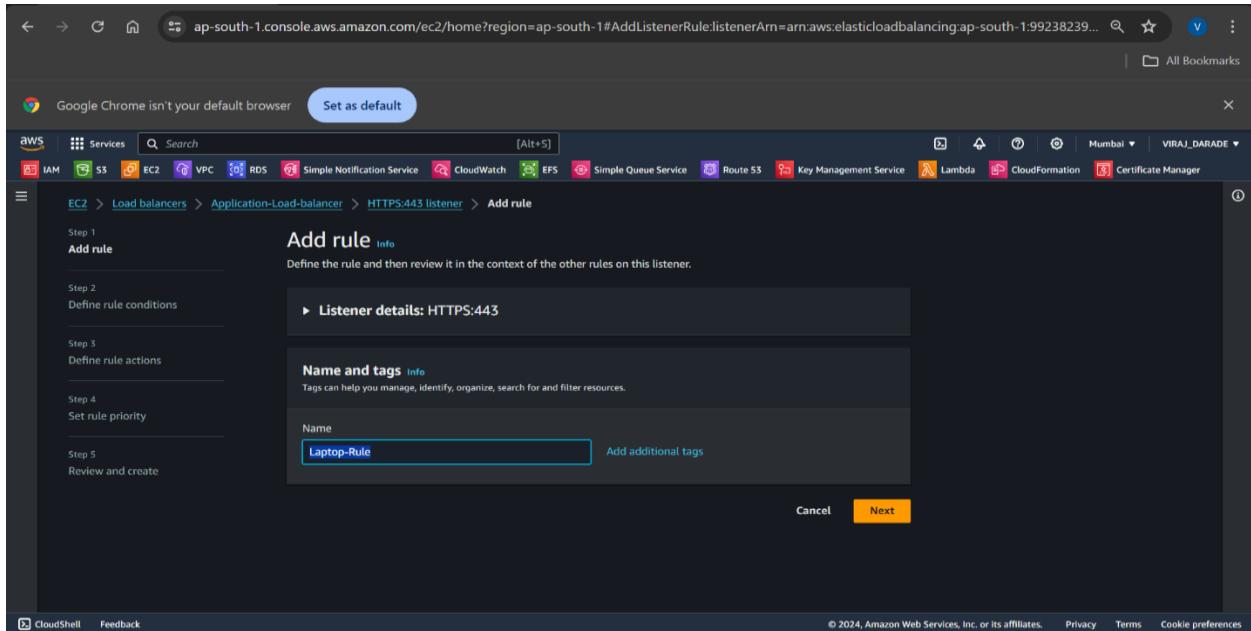
Protocol:Port: Default action: Forward to target group: Home-TG (100%)

Manage rules ▲ Manage listener ▾ Add rule Edit rules Reprioritize rules

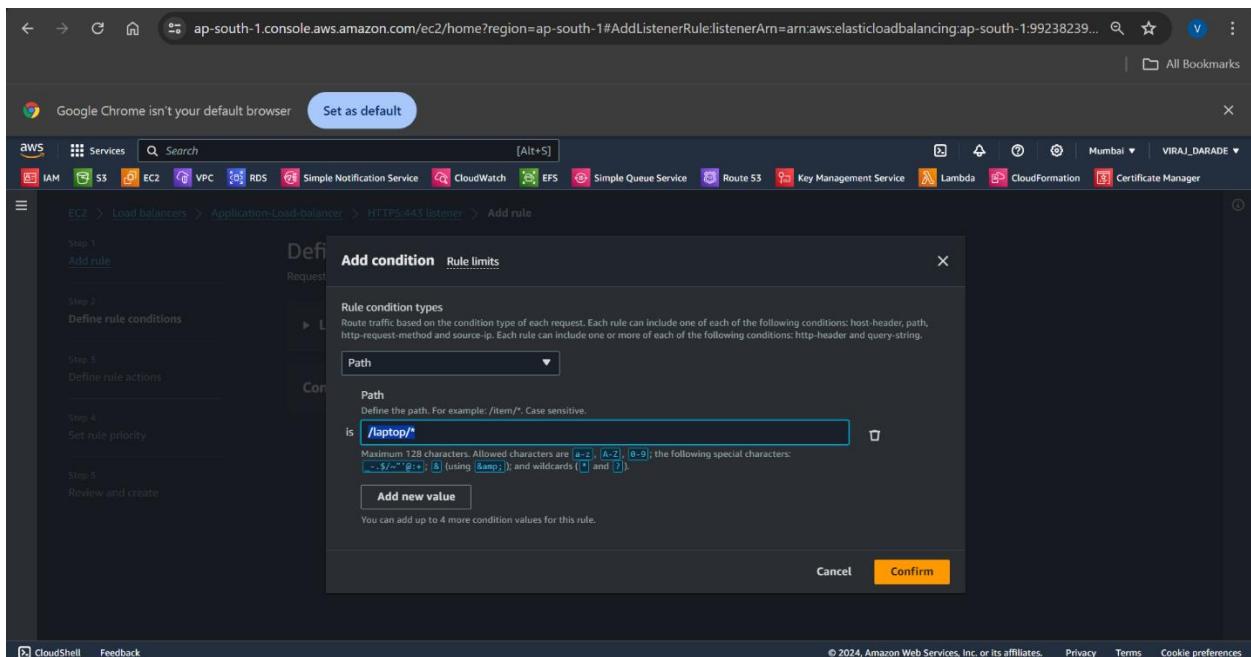
Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate
HTTPS:443	Forward to target group: Home-TG (100%)	1 rule	arn:aws:elasticloadbalancing:ap-south-1:992382392415:loadbalancer/app/Application-Load-balancer/22de503898a58524	ELBSecurityPolicy-TLS13-1-2...	viraaj.online (Certificate ID: ...)

At the bottom right of the modal, there are links for 'Privacy', 'Terms', and 'Cookie preferences'.

- ❖ Name : **Laptop_Rule**
- ❖ Click On **Next Button**



- ❖ Add Condition :
- ❖ Rule Condition Type : Select **Path**
- ❖ Path : **/Laptop/***
- ❖ Click On **Confirm Button**

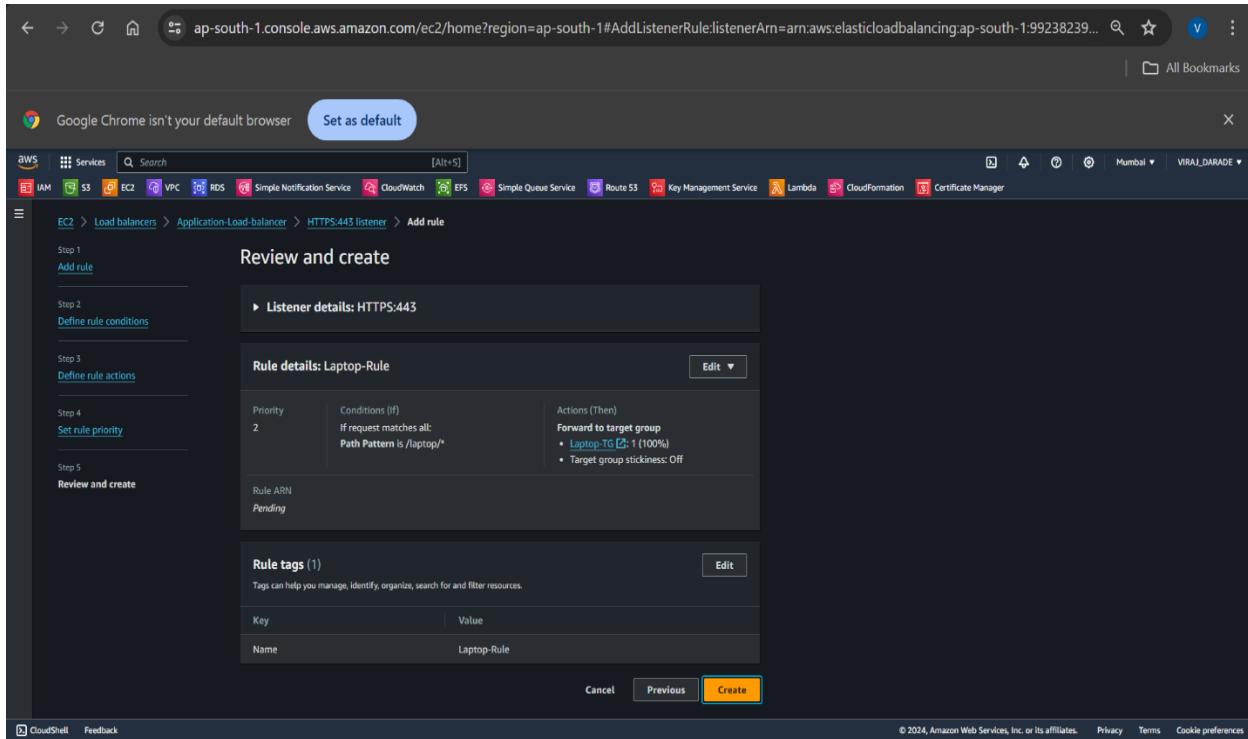


The screenshot shows the AWS CloudFront console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#AddListenerRule:listenerArn=arn:aws:elasticloadbalancing:ap-south-1:99238239.... The page is titled "Define rule conditions" and shows a "Conditions (1)" section with a "Path (1)" condition. The condition is defined as "Path is /laptop/*". The sidebar on the left lists steps: Step 1 (Add rule), Step 2 (Define rule conditions), Step 3 (Define rule actions), Step 4 (Set rule priority), and Step 5 (Review and create). The "Define rule actions" step is currently selected.

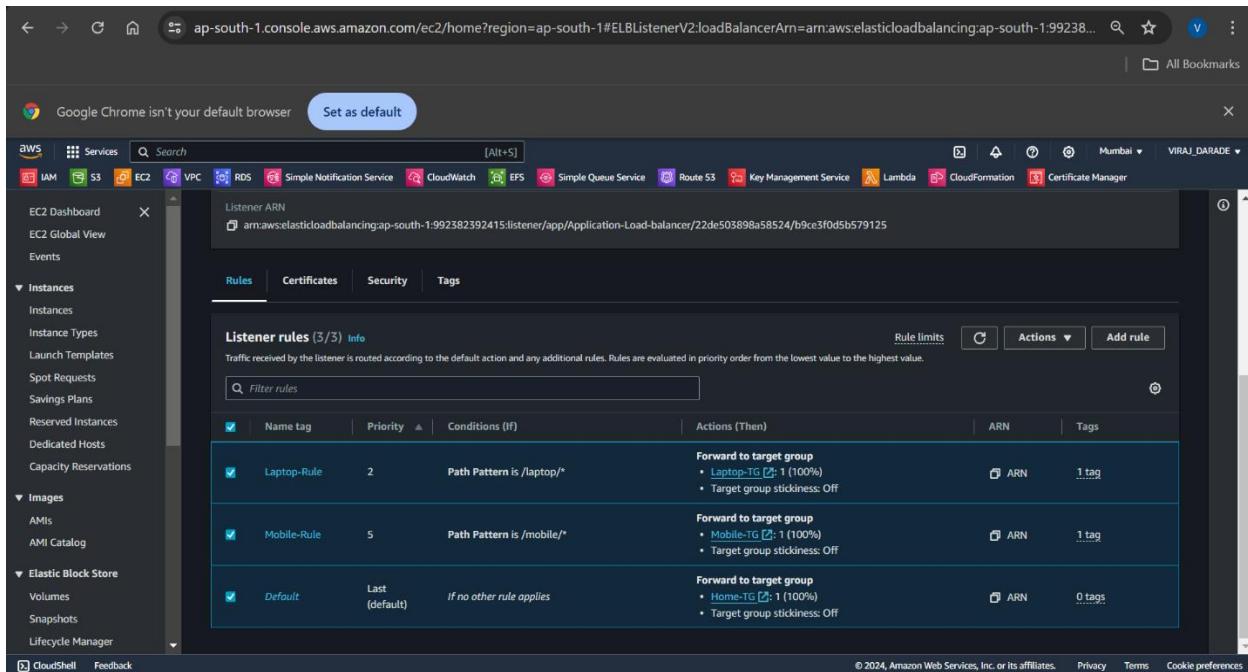
- ❖ Routing Action : Select **Forward To Target Groups**
- ❖ Target Group : Select **Laptop-TG**

The screenshot shows the AWS CloudFront console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#AddListenerRule:listenerArn=arn:aws:elasticloadbalancing:ap-south-1:99238239.... The page is titled "Actions" and shows the "Action types" section with "Authentication" and "Forward to target groups" selected. Under "Forward to target groups", a "Target group" section is shown with "Laptop-TG" selected. The sidebar on the left lists steps: Step 1 (Add rule), Step 2 (Define rule conditions), Step 3 (Define rule actions), Step 4 (Set rule priority), and Step 5 (Review and create). The "Define rule actions" step is currently selected.

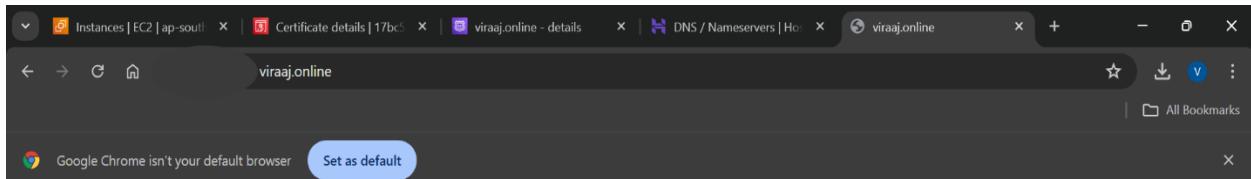
❖ Review & Click On Create Button



- ❖ Repeat Same Process For Mobile Rule
- ❖ Only Change Path & Target Group
- ❖ Here We Can See, Our Rules Will Be Created (**Mobile-Rule, Laptop-Rule**)



- ❖ Hit Our **Domain Name To Any Web Browser**
- ❖ Domain Name : **viraaj.online**
- ❖ Here We Can See, Our **Home Webpage**
- ❖ When We Refresh Our **Webpage** We Also Displays **Laptop & Mobile Webpage** Randomly With Different Ips (Round Robin)



HOME PAGE ip-172-31-13-179.ap-south-1.compute.internal

-
- ❖ By Following These Steps, You Will Have A Secure, Scalable, And Highly Available Website Hosted On AWS, Leveraging ALB, SSL Certificates For HTTPS, And Auto Scaling To Handle Varying Traffic Loads Efficiently.