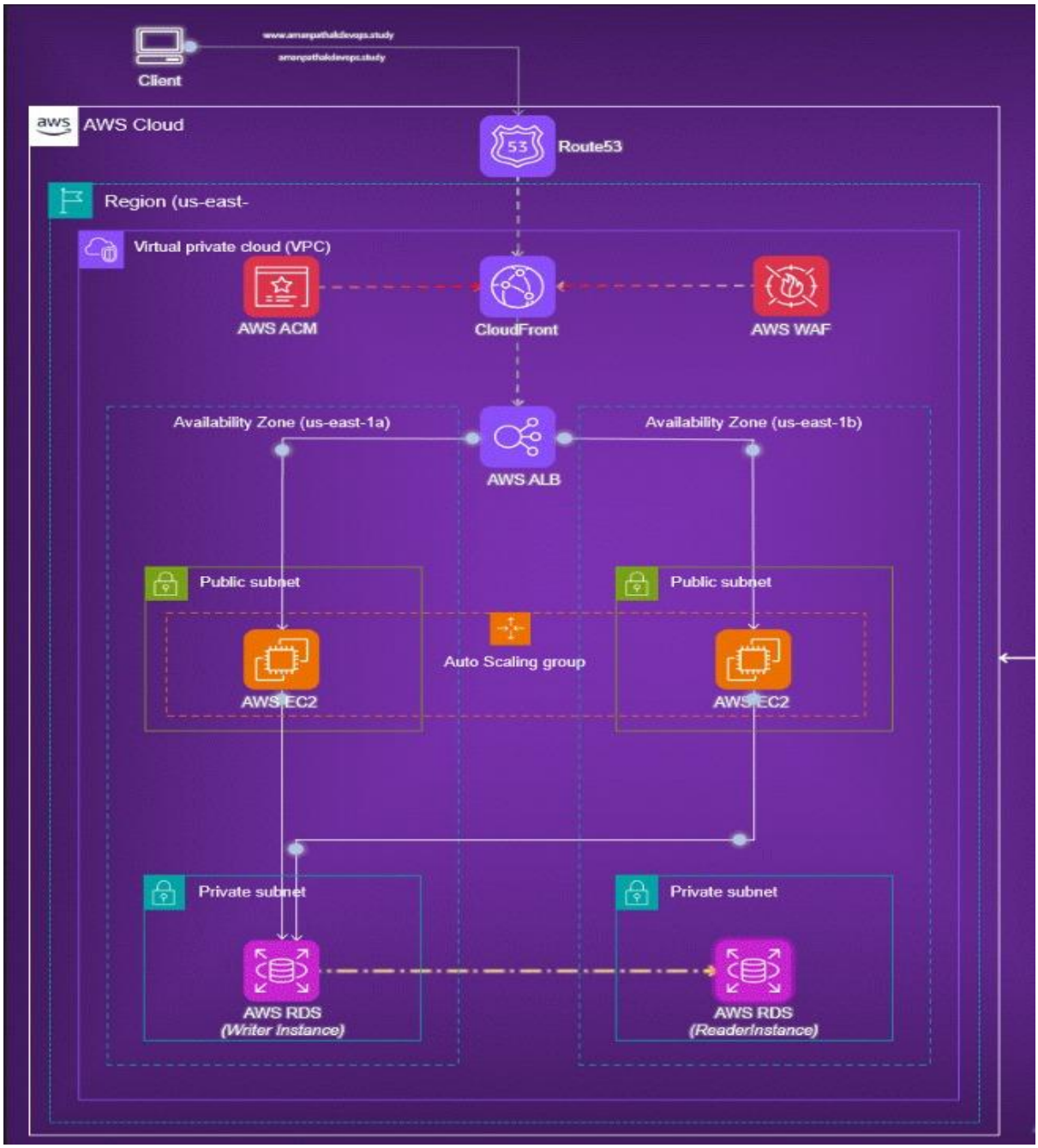
Project -2

2-TIRE STATIC WEB HOSTING

NAME: DASARI SIVA SANKAR

MAIL.ID: Dasarisivasankar6@gmail.com

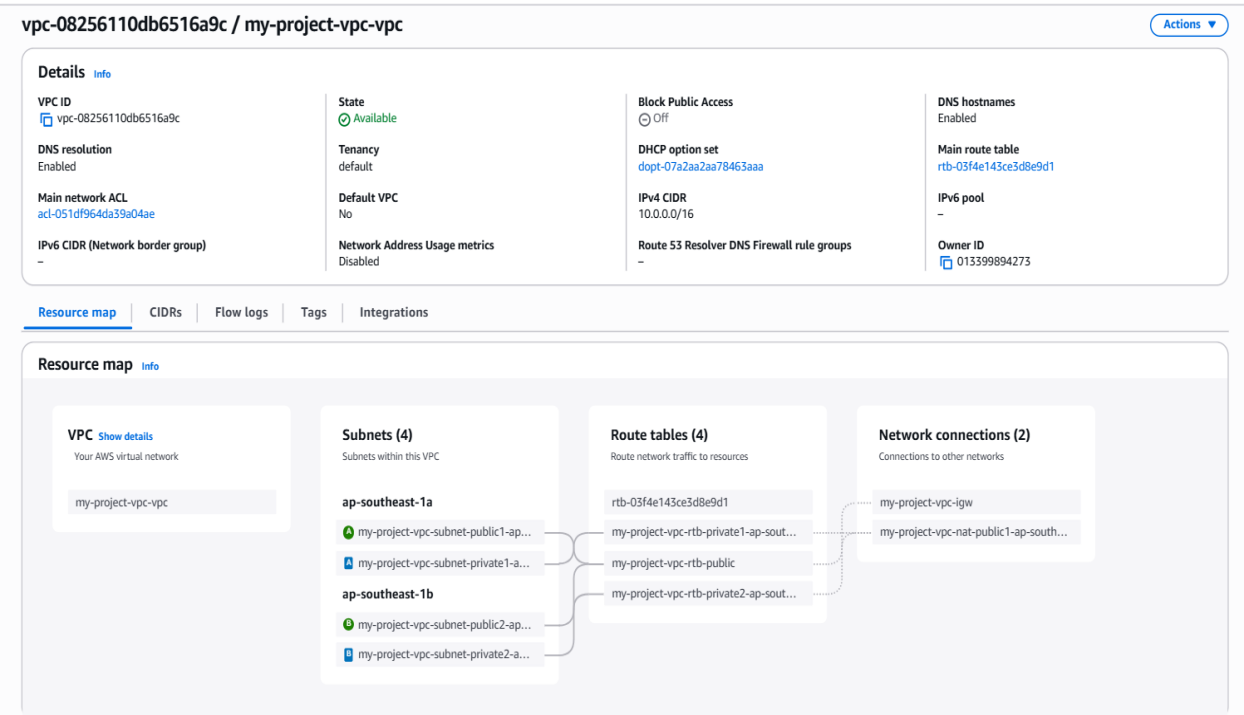


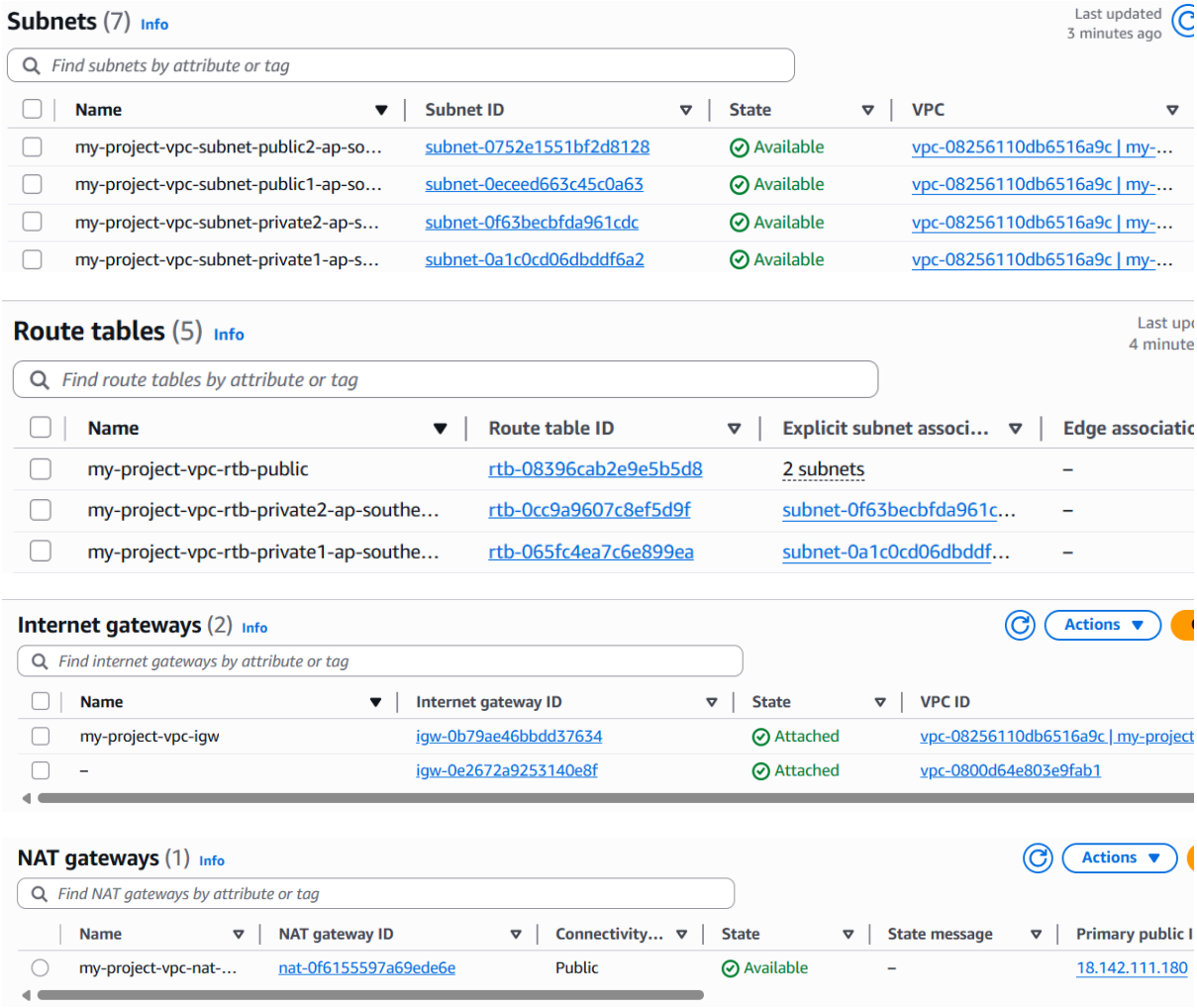
1.**Set Up the VPC**

- Go to **VPC** → Launch **VPC MORE**:
    - Create a **VPC and more**.
    - Add **2 public subnets** (e.g., us-west-1a, us-west-1c).
    - Add **2 private subnets**.
    - Enable **DNS resolution** and **DNS hostnames**.
    - **Create Internet Gateway & Route Tables.**
    - Attach an **Internet Gateway (IGW)** to your VPC.

Create route tables:

- Public subnets → Route to IGW.
- Private subnets → No direct internet access.





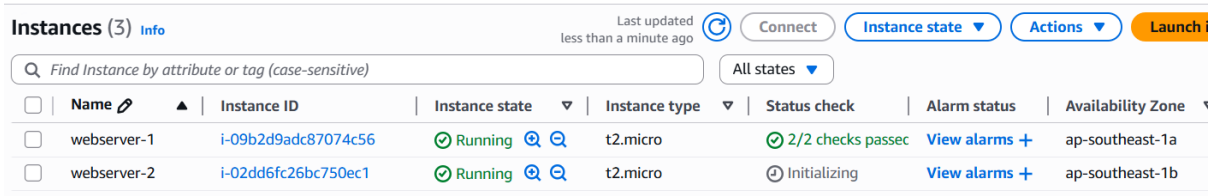
2.Launch EC2 Instances

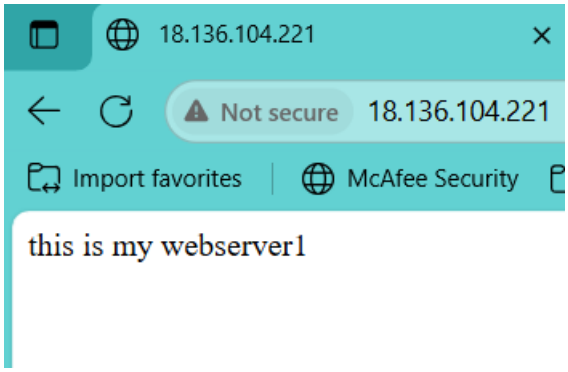
- Go to **EC2** → Launch Instances:
    - Click **"Launch Instances"**
    - Named them webserver-1 and webserver-2
    - **Key Pair**: Create or select an existing one

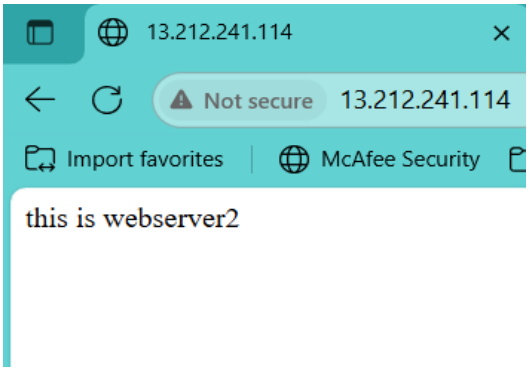
Network Settings:

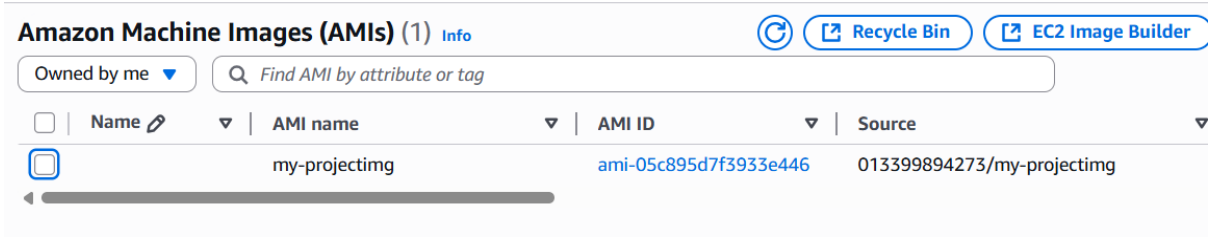
- Allow **SSH (22)** and **HTTP (80)**
    - Connect via SSH
    - Run this on both instances
    - Install Web Server

- sudo -I && apt update -y
- apt install apache2
- systemctl start apache2
- systemctl status apache2
- After create **Amazon Machine Images (AMI)**





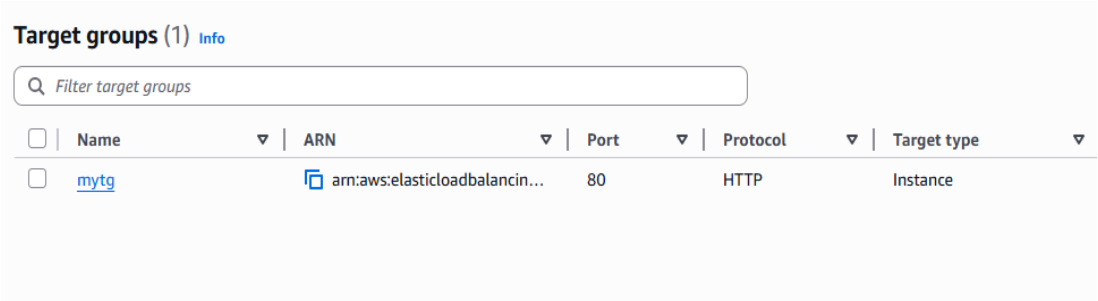


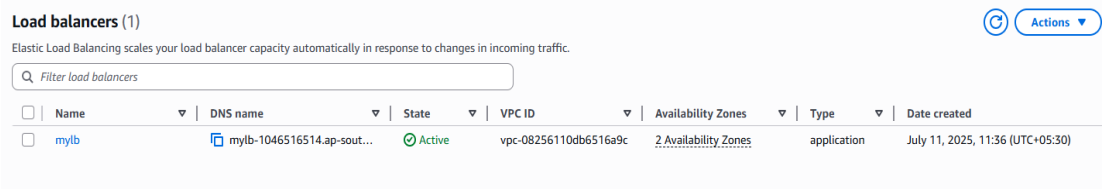


3. Create an Application Load Balancer

- **Configure Target Group**
- **Target group name**: my-target-group
- **Target type**: Instance
- **Availability Zones**: Select your **VPC**, and then select **two public subnets**
- Select an **existing security group**
- Select Load Balancer Type

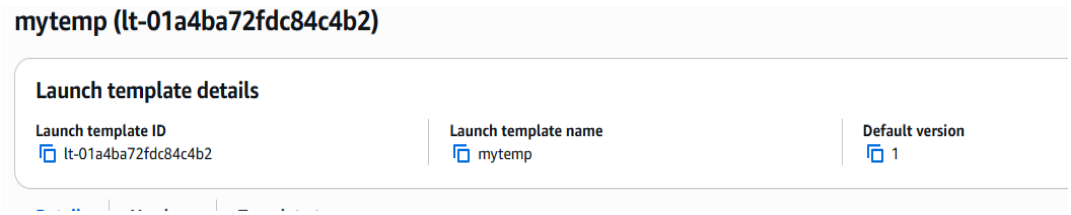
- Choose **Application Load Balancer**





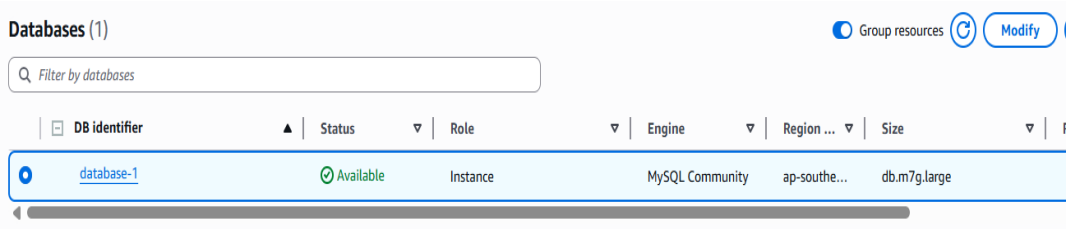
4.Create Auto Scaling

- Navigate to **EC2 Dashboard** → **Launch Templates**.
- Click **Create launch template**.
- **Key pair**: Select an existing key (for SSH access).
- **Security Group**: Allow HTTP (80) and/or SSH (22).
- Open Auto Scaling Groups
- Go to **EC2 Dashboard** → Scroll down → Click **Auto Scaling Groups**.
- Click **Create Auto Scaling group**.
- Configure Network
- **VPC**: Choose your VPC.
- **Availability Zones and subnets**: Select **at least 2 public subnets**
- You can attach an **existing ALB** here
- Select your **Target Group**
- **Minimum capacity**: 1
- **Minimum capacity**: 4



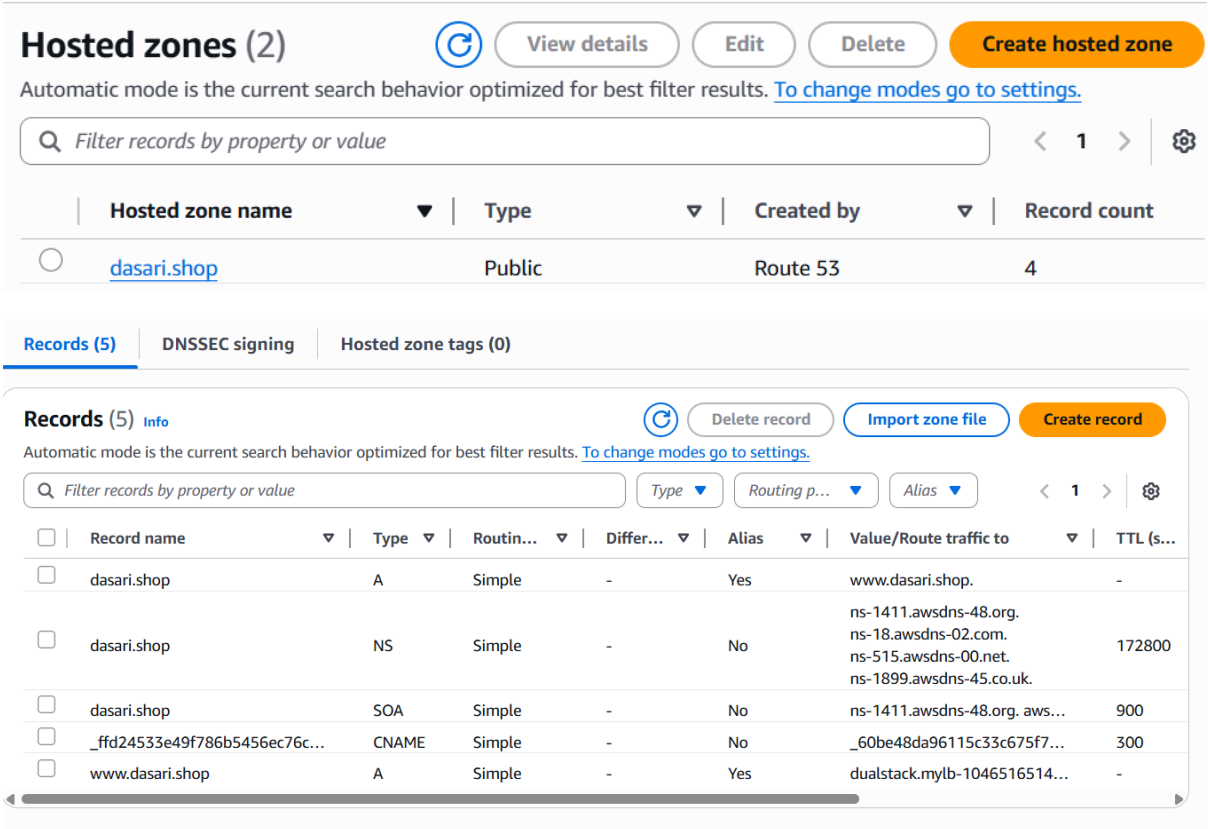
4. Now Create RDS

- Go to AWS RDS Console
- Click **Create Database**
- Choose **Standard Create**
- Choose **MySQL**
- **Multi-Az DB instance identifier**: database-1
- **Master username**: e.g., admin
- **Set a strong password**
- For production: Choose based on your needs
- **(VPC)**: Choose an existing VPC
- Add rule for **inbound access on port 3306** (MySQL)
- Click **Create database**



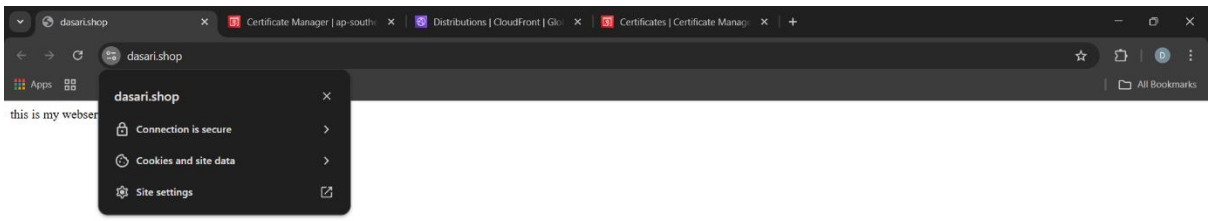
5. Using Route 53 in AWS

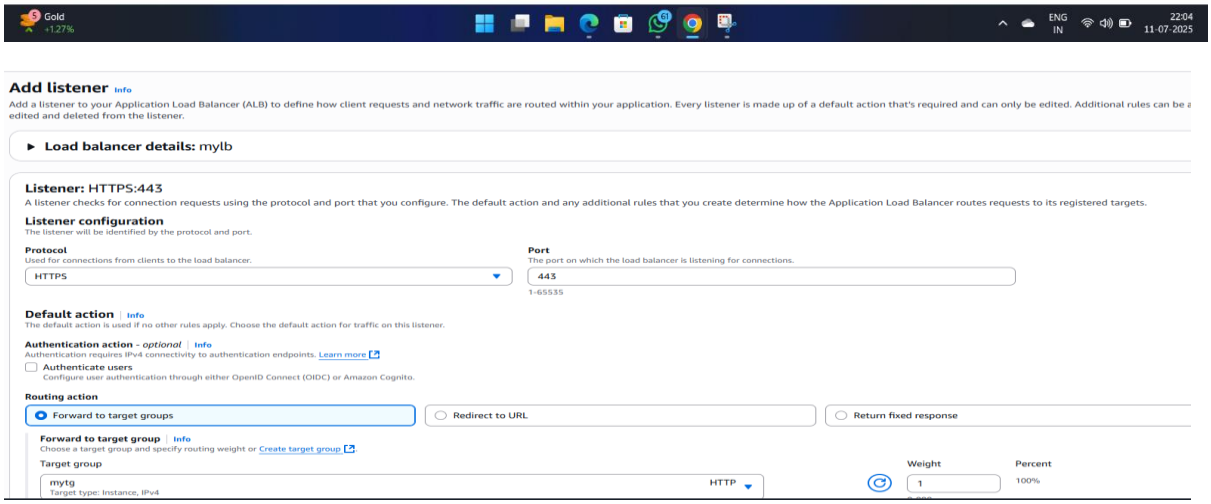
- Click on Route 53 to open the service
- Create a Hosted Zone
- Click **Create hosted zone**.
- Enter your **domain name**
- Choose **Public hosted zone**
- Update Name Servers
    1. Go to your domain registrar (e.g., GoDaddy)
    2. replace default Name Servers with the **NS records from Route 53**
    - Click **Create record**
    - Record name: Leave blank or add www if needed
    - Alias Load Balancer



6: Secure the Domain

- Initially, the domain used HTTP and was not secure.
- To make it secure: o Set up a CloudFront Distribution using the certificate.
- Added an HTTPS listener to the Load Balancer.
- Verified the domain was accessible securely (HTTPS) using the CloudFront domain






```
eading /usr/share/mecab/dic/ipadic/Noun.org.csv ... 16668
eading /usr/share/mecab/dic/ipadic/Suffix.csv ... 1393
eading /usr/share/mecab/dic/ipadic/Postp.csv ... 146
eading /usr/share/mecab/dic/ipadic/Noun.nai.csv ... 42
eading /usr/share/mecab/dic/ipadic/Symbol.csv ... 208
eading /usr/share/mecab/dic/ipadic/Adverb.csv ... 3032
eading /usr/share/mecab/dic/ipadic/Noun.demonst.csv ... 120
eading /usr/share/mecab/dic/ipadic/Adnominal.csv ... 135
eading /usr/share/mecab/dic/ipadic/Postp-col.csv ... 91
eading /usr/share/mecab/dic/ipadic/Interjection.csv ... 252
eading /usr/share/mecab/dic/ipadic/Adj.csv ... 27210
eading /usr/share/mecab/dic/ipadic/Noun.others.csv ... 151
eading /usr/share/mecab/dic/ipadic/Auxil.csv ... 199
eading /usr/share/mecab/dic/ipadic/Noun.proper.csv ... 27328
emitting double-array: 100% |###########################################|
eading /usr/share/mecab/dic/ipadic/matrix.def ... 1316x1316
emitting matrix      : 100% |###########################################|

done!
pdate-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode
Setting up libhtml-parser-perl:amd64 (3.81-1build3) ...
Setting up libhttp-message-perl (6.45-1ubuntu1) ...
Setting up mysql-server (8.0.42-0ubuntu0.24.04.1) ...
Setting up libcgi-pm-perl (4.63-1) ...
Setting up libhtml-template-perl (2.97-2) ...
Setting up libcgi-fast-perl (1:2.17-1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-20-0-1-19:~#
```

```
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database Gowri;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| Gowri              |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use Gowri;
Database changed
mysql> CREATE TABLE Gowri ( ID int NOT NULL,
    -> LastName varchar(255) NOT NULL, FirstName varchar(255),
    -> Age int,
    -> PRIMARY KEY (ID)
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO Gowri (ID, LastName, FirstName, Age)
    -> VALUES (432, 'shiva', 'sankar', 23);
Query OK, 1 row affected (0.00 sec)

mysql> select * from Gowri;
+-----+----------+-----------+------+
| ID  | LastName | FirstName | Age  |
+-----+----------+-----------+------+
| 432 | shiva    | sankar    |   23 |
+-----+----------+-----------+------+
1 row in set (0.00 sec)

mysql>
```

Final Conclusion:

- Launched and configured two EC2 instances with Apache.
- Set up load balancing using Target Group and Application Load Balancer.
- Registered a domain and configured Route 53 records.
- Mapped domain name servers to Route 53 for DNS resolution.
- Made the domain secure using ACM and CloudFront.
- The website was successfully deployed and is accessible securely through the domain.