1. Configure Classic Load balancer.

**Create a Load Balancer**

* Navigate to **Load Balancers** under **EC2**.
* Click **Create Load Balancer**.
* Choose **Classic Load Balancer**.
* Click **Continue**.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Create 2 public instances with different regions subnet

Write the script to download httpd in both ec2 and create index.html files

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

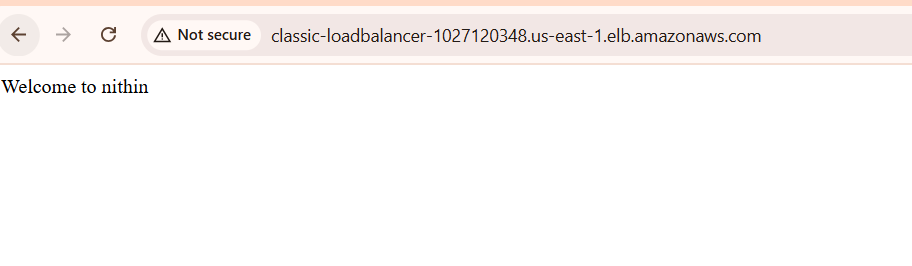
AI-generated content may be incorrect.

Broswe:

classic-loadbalancer-1027120348.us-east-1.elb.amazonaws.com

A screenshot of a computer

AI-generated content may be incorrect.



1. Configure Application Load balancer.

**Step 1: Prepare EC2 Instances**

Make sure you have:

* At least **2 EC2 instances** running in **2 different Availability Zones**.
* Apache or Nginx installed (serving a web page).
* **Security group** allowing:
  + **Inbound HTTP (port 80)**
  + **Outbound internet access**

A screenshot of a computer

AI-generated content may be incorrect.

**✅ Step 2: Create a Target Group**

Go to **EC2 Dashboard > Target Groups**.

Click **Create target group**.

Select:

* + **Target type**: Instances
  + **Protocol**: HTTP
  + **Port**: 80

Give it a name (e.g., my-target-group).

Choose **VPC**.

**Health check path**: /

Click **Next**.

Select the EC2 instances to include.

Click **Include as pending below**, then **Create target group**.

A screenshot of a computer

AI-generated content may be incorrect.

A close-up of a computer screen

AI-generated content may be incorrect.

**Step 3: Create the Application Load Balancer**

Go to **EC2 Dashboard > Load Balancers**.

Click **Create Load Balancer** → Choose **Application Load Balancer**.

Set the following:

* + **Name**: my-alb
  + **Scheme**: Internet-facing
  + **IP type**: IPv4
  + **Listeners**: HTTP (port 80)

**Availability Zones**:

* + Select at least 2 AZs
  + Select the appropriate **subnets**

Click **Next: Configure Security Settings** → Skip this (HTTP only).

Click **Next: Configure Security Groups**:

* + Use an existing group or create one that allows **HTTP (port 80) inbound**.

Click **Next: Configure Routing**:

* + Choose the **target group** you created earlier.
  + Click **Next** → **Create Load Balancer**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**tep 4: Test the ALB**

1. Go to **Load Balancers** → Select your ALB.
2. Copy the **DNS name** of the ALB
3. Open your browser and go to:

http://application-loadbalancer-711334277.us-east-1.elb.amazonaws.com/

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1. Configure Network Load balancer.

**Step 1: Prepare EC2 Instances**

Make sure you have **2 or more EC2 instances** running in **different Availability Zones**, with a service listening on a **TCP port** (commonly port 80 or 8080).

A screenshot of a computer

AI-generated content may be incorrect.

**Step 2: Create a Target Group**

1. Go to **EC2 Dashboard > Target Groups**.
2. Click **Create target group**.
3. Select:
   * **Target type**: Instances
   * **Protocol**: TCP
   * **Port**: 80 (or the port your app uses)
   * **Target group name**: e.g., nlb-target-group
4. Choose your **VPC**.
5. Leave **health check** protocol as TCP.
6. Click **Next**.
7. Select your EC2 instances.
8. Click **Include as pending** → then **Create target group**.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A close-up of a computer screen

AI-generated content may be incorrect.

**Step 3: Create the Network Load Balancer**

1. Go to **EC2 Dashboard > Load Balancers**.
2. Click **Create Load Balancer** → Choose **Network Load Balancer**.
3. Fill in the following:
   * **Name**: my-nlb
   * **Scheme**: Internet-facing (for public access)
   * **IP address type**: IPv4
4. **Listeners**:
   * Protocol: TCP
   * Port: 80 (or your app’s port)
   * Forward to: your target group (created above)
5. **Availability Zones**:
   * Select at least 2 AZs and their subnets.
6. Click **Create Load Balancer**

A computer screen with text boxes

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 4: Test Your NLB**

1. Go to **Load Balancers** → find your NLB.
2. Copy its **DNS name**

A screenshot of a computer

AI-generated content may be incorrect.

1. Attach SSL for application load balancer.

**🔹 Step 1: Get an SSL Certificate from ACM**

Go to the **AWS Management Console** → **Certificate Manager (ACM)**.

Click **Request a certificate**.

Choose **Request a public certificate** (for public domains).

Enter your domain name (e.g., www.example.com).

Choose a validation method:

* + DNS validation (easier with Route 53)
  + Email validation

Complete the validation process.

Once the certificate status is **"Issued"**, move to the next step

Map Applciation load balancer to R53.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**🔹 Step 2: Go to Your ALB in EC2 Console**

1. Open the **EC2 Console** → **Load Balancers** (under **Load Balancing**).
2. Select your **Application Load Balancer**.
3. Go to the **Listeners** tab.
4. Click **Add listener**.

**🔹 Step 3: Create HTTPS Listener on Port 443**

* **Protocol:** HTTPS
* **Port:** 443
* **Default Action:** Forward to your **Target Group** (e.g., web servers)

Under **SSL certificate**:

* Choose **"From ACM"**
* Select the certificate you created in Step 1

Click **Add** or **Save**.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 5: Test It**

Open a browser and visit:

http://sripriya.shop/

A screenshot of a computer

AI-generated content may be incorrect.

5) Map Applciation load balancer to R53.

**🔹 Step 1: Get Your ALB DNS Name**

Go to the **EC2 Console** → **Load Balancers**.

Select your **Application Load Balancer**.

Copy the **DNS name**,

A screenshot of a computer

AI-generated content may be incorrect.

**🔹 Step 2: Go to Route 53 Hosted Zone**

Open the **Route 53 Console**.

Select **Hosted Zones** from the left sidebar.

Choose your hosted zone (e.g., example.com).

**🔹 Step 3: Create Alias Record for Your Domain**

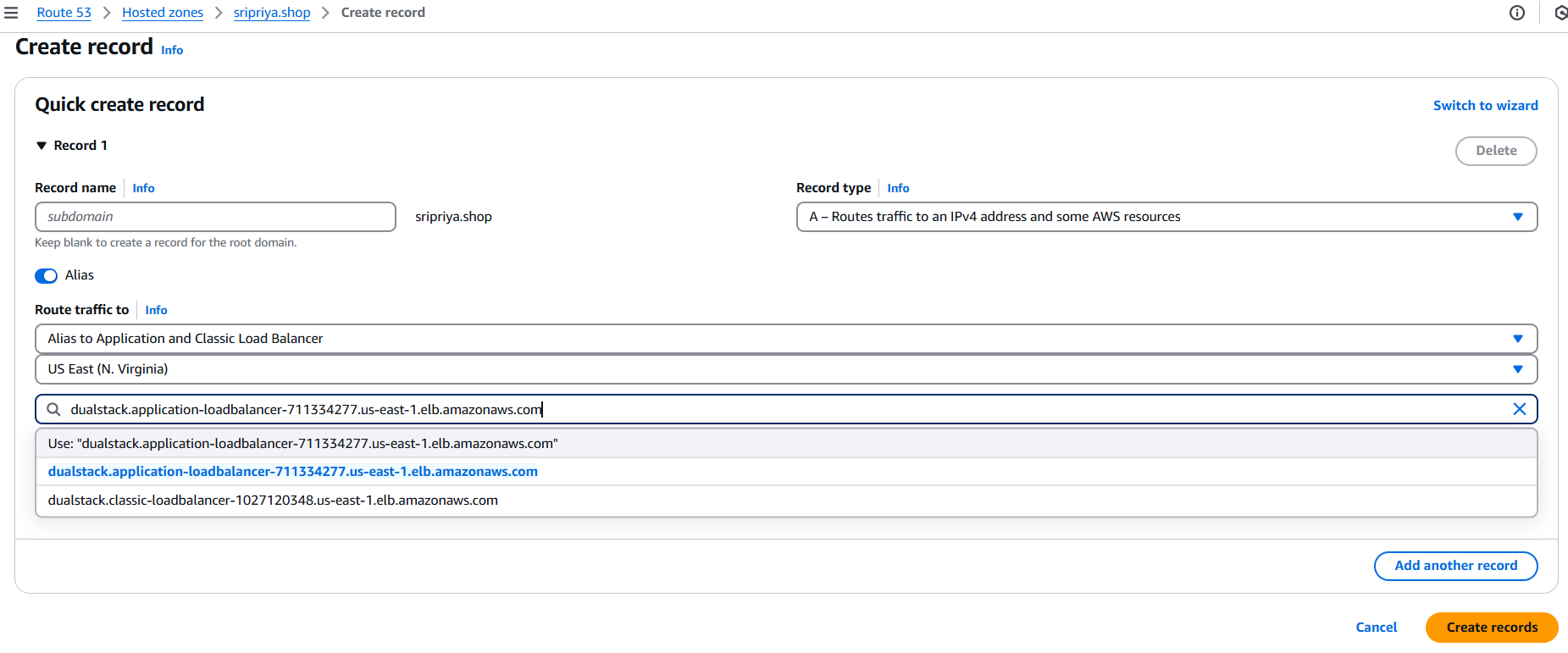
Click **Create Record**.

Fill in the record:

* + **Record type:** A (IPv4 address)
  + **Alias:** Yes
  + **Alias target:** Click in the field and choose your ALB from the dropdown.

✅ AWS auto-suggests your ALB DNS name if it's in the same region.

1. Click **Create records**.



🔹 Step 5: Test in Browser

A screen shot of a message

AI-generated content may be incorrect.

6)Push the application load balancer logs to s3.

**Step 1: Create an S3 Bucket (if not already created)**

Go to **S3 Console** → Click **Create bucket**

* Bucket name: my-alb-logs-bucket (or your preferred name)
* Region: Same as your Load Balancer

A screenshot of a computer

AI-generated content may be incorrect.

**🔹 Step 2: Add S3 Bucket Policy**

Click your S3 bucket → **Permissions** → **Bucket policy**

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "AWSALBLoggingPermissions",

"Effect": "Allow",

"Principal": {

"Service": "logdelivery.elasticloadbalancing.amazonaws.com"

},

"Action": "s3:PutObject",

"Resource": "arn:aws:s3:::my-alb-logs-bucke/\*"

}

]

}

A screenshot of a computer

AI-generated content may be incorrect.

**🔹 Step 3: Enable Access Logs on ALB**

1. Go to the **EC2 Console** → **Load Balancers**
2. Select your **Application Load Balancer**
3. Go to **Attributes** tab → Click **Edit attributes**
4. Under **Access logs**:
   * ✅ Enable Access logs
   * S3 location: my-alb-logs-bucket
   * S3 prefix: alb-logs/ (optional but recommended)
5. Click **Save Changes**

A screenshot of a computer

AI-generated content may be incorrect.

Step 4: Wait for Logs

A screenshot of a computer

AI-generated content may be incorrect.