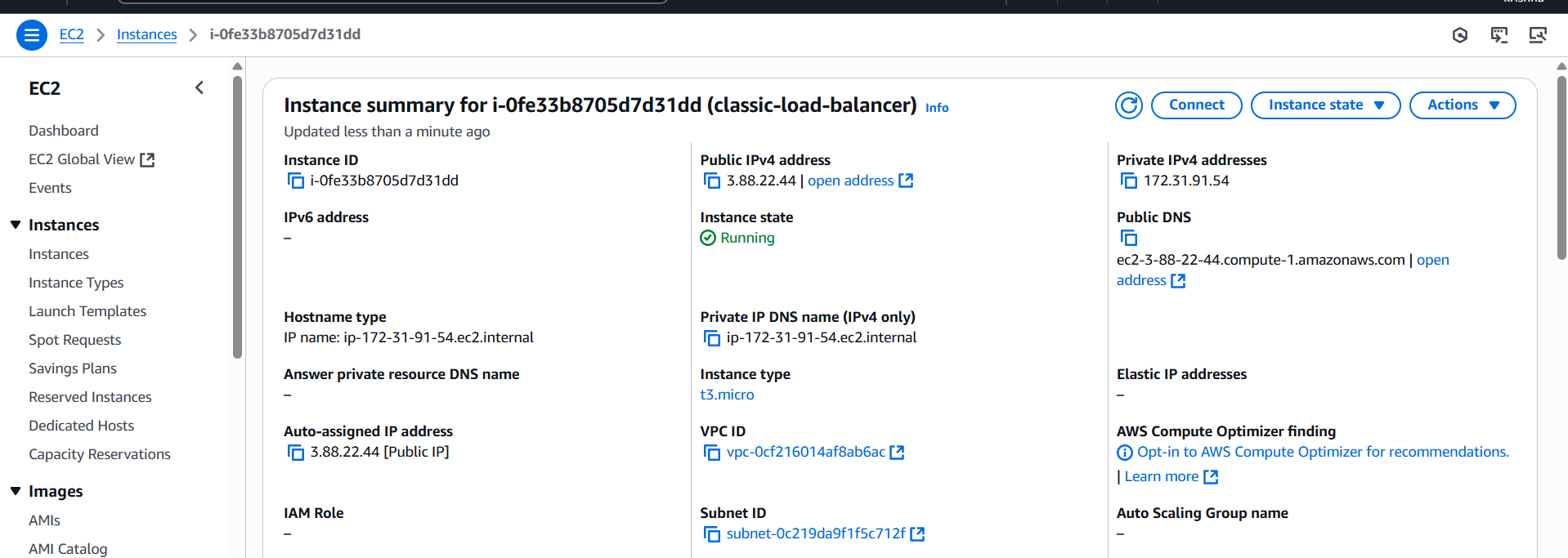
1) Configure Classic Load balancer.

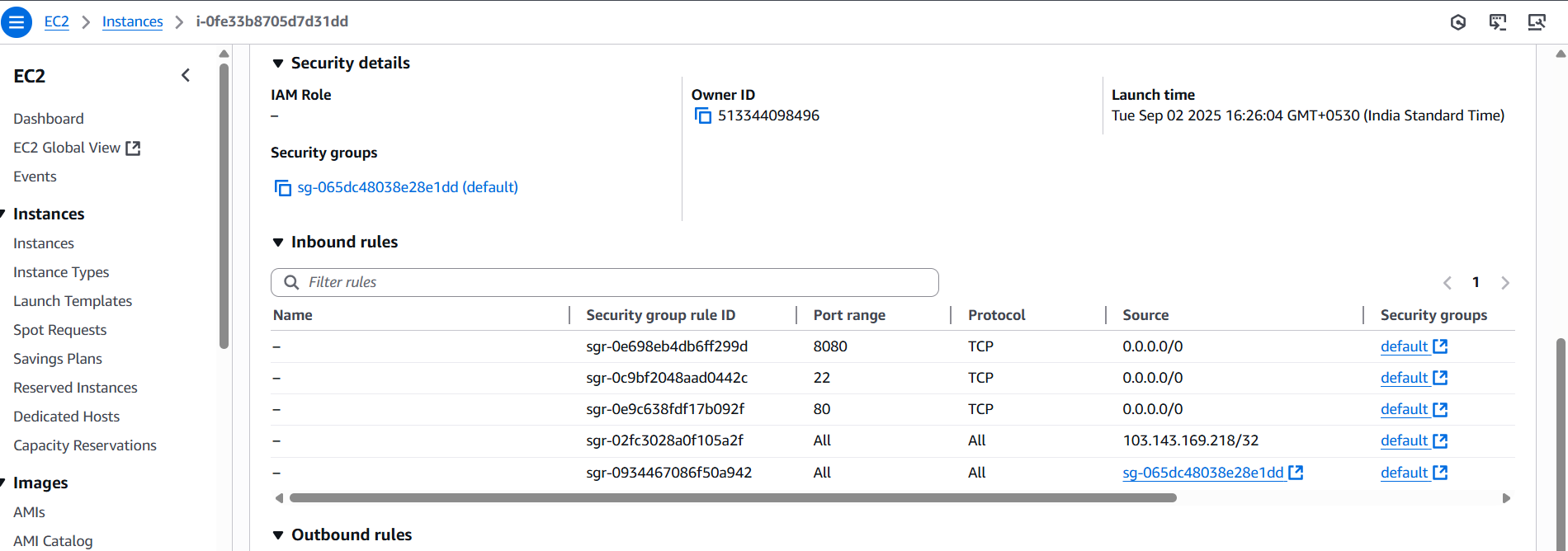
Step 1: Create a Ec2 Instance:

**1. Go to EC2 Dashboard**

* Open AWS Console.
* Go to **EC2 > Instance**.
* Choose **Launch Instance**.

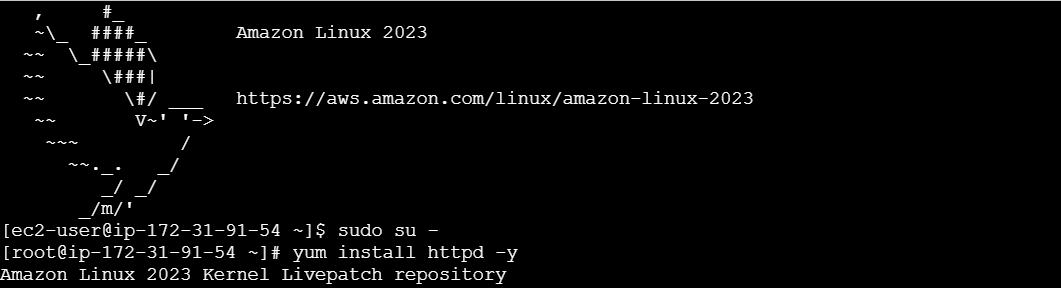


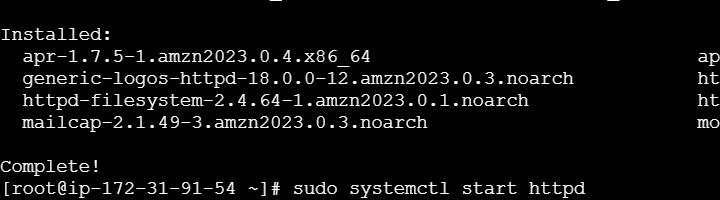
Check for the security group whether it is having to http port 80, if not give access by edit the inbound rules



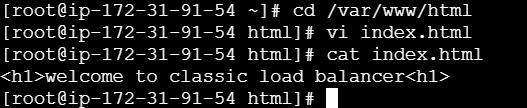
2. Install httpd:

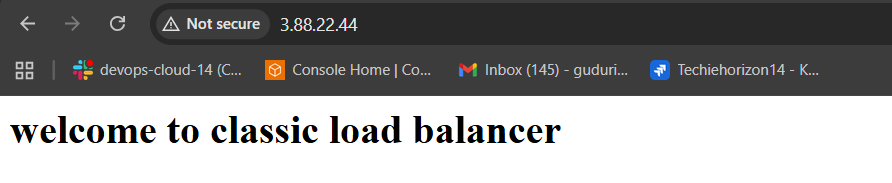
* Connect to the instance
* Use yum install httpd -y
* Sudo systemctl start httpd





Add an index.html file

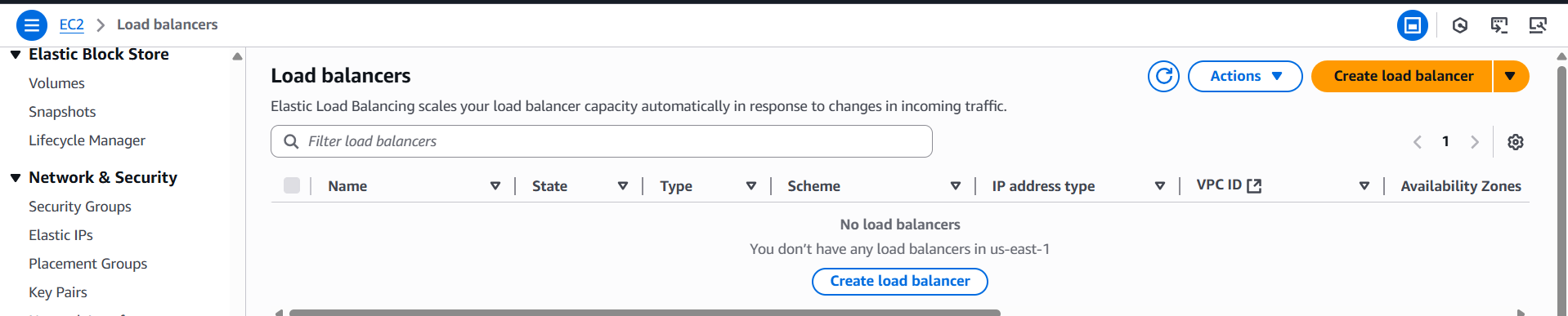




Step 2: Configure Classic Load Balancer

**1. Go to EC2 Dashboard**

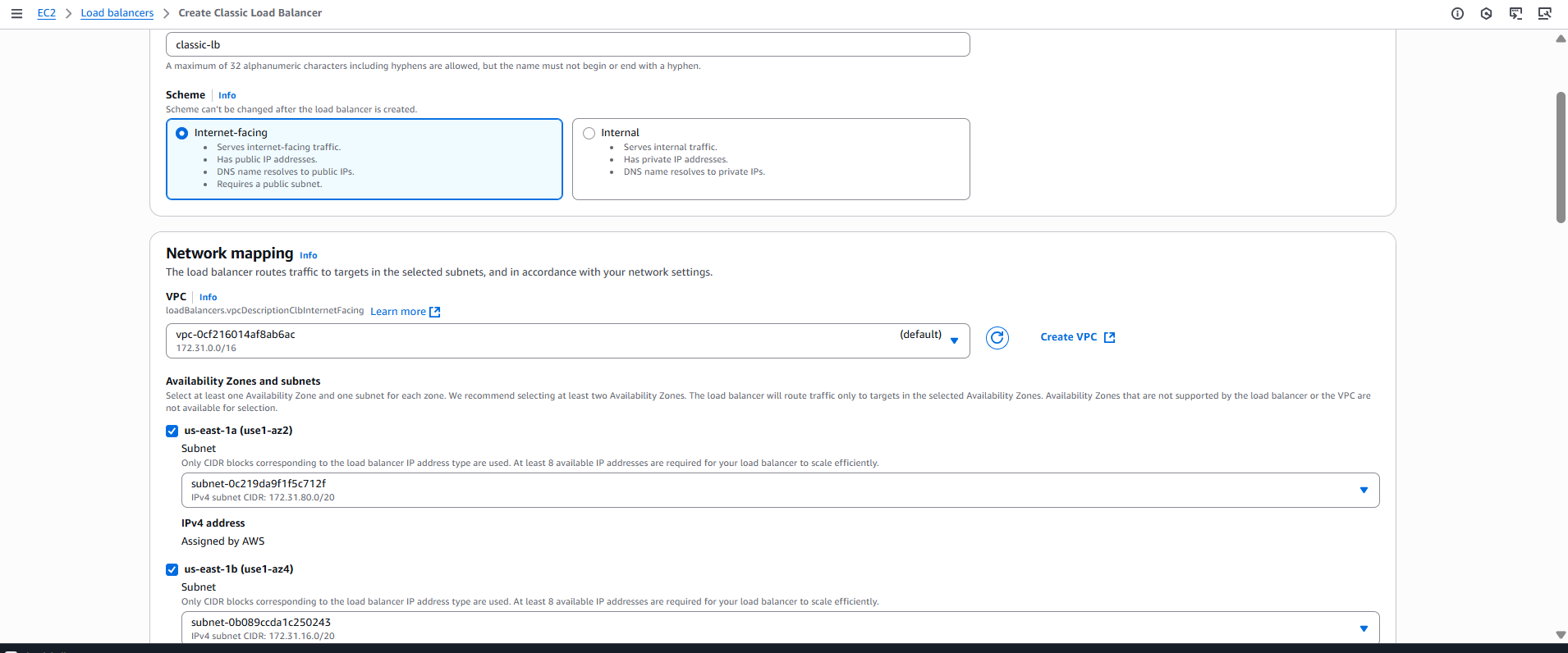
* Open AWS Console.
* Go to **EC2 > Load Balancers**.
* Click **“Create Load Balancer”**.
* Choose **Classic Load Balancer**.

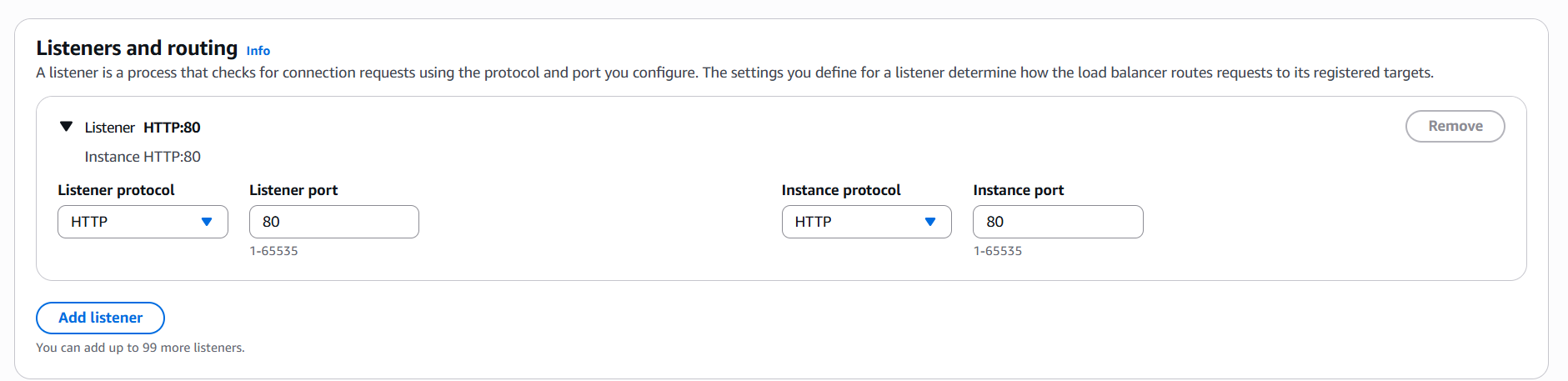


**2. Set Load Balancer Name & Listener**

* Name it (e.g., classic-lb).
* Listener: HTTP 80 → HTTP 80 (default).
* Choose **VPC** and **Availability Zones** (where your EC2s are).

Click **Next**.

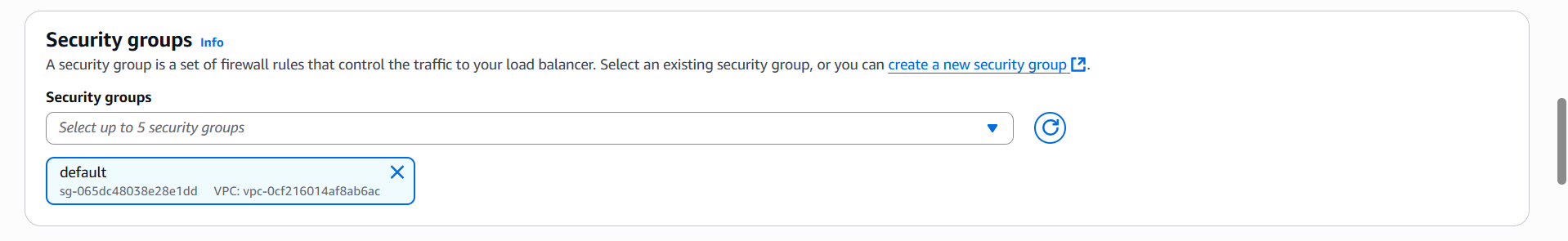




**3. Security Group**

* Select or create a security group.
* Allow **HTTP (port 80)** and **health check port**.

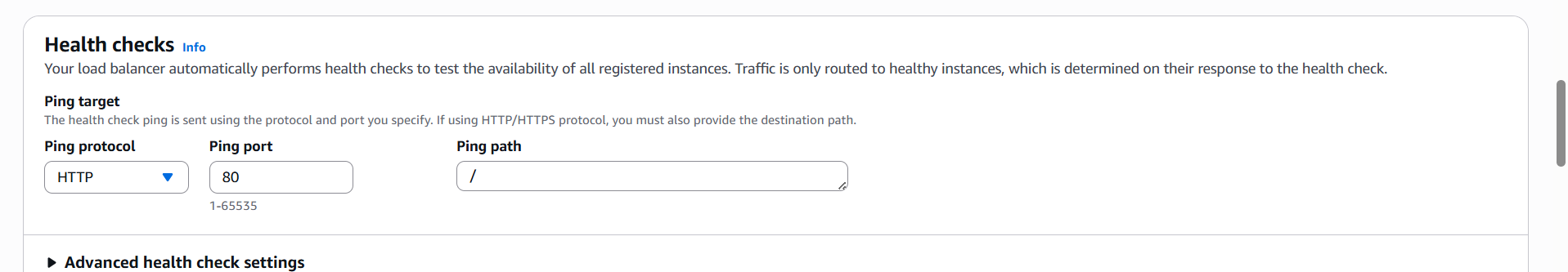
Click **Next**.



**4. Health Check Settings**

* Protocol: HTTP
* Port: 80
* Path: /
* Leave other settings as default (or adjust if needed).

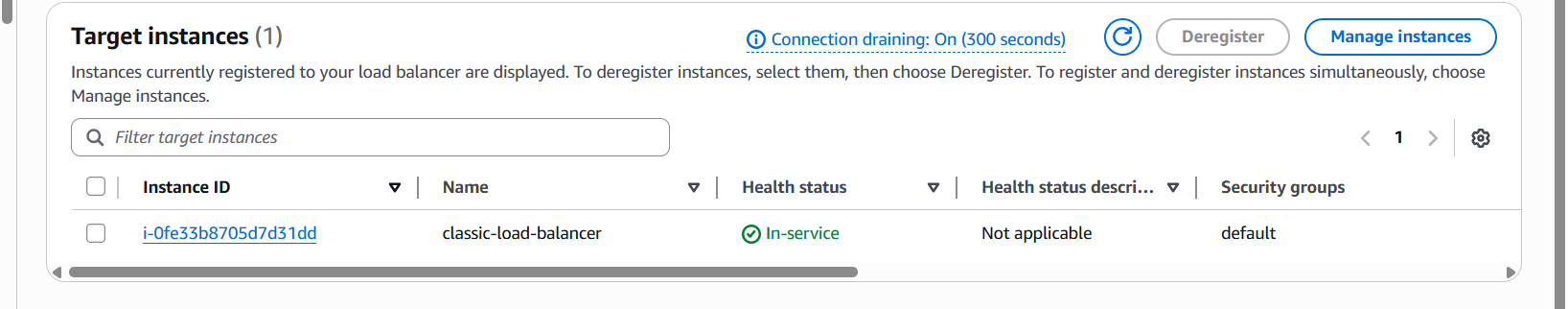
Click **Next**.



**5. Register EC2 Instances**

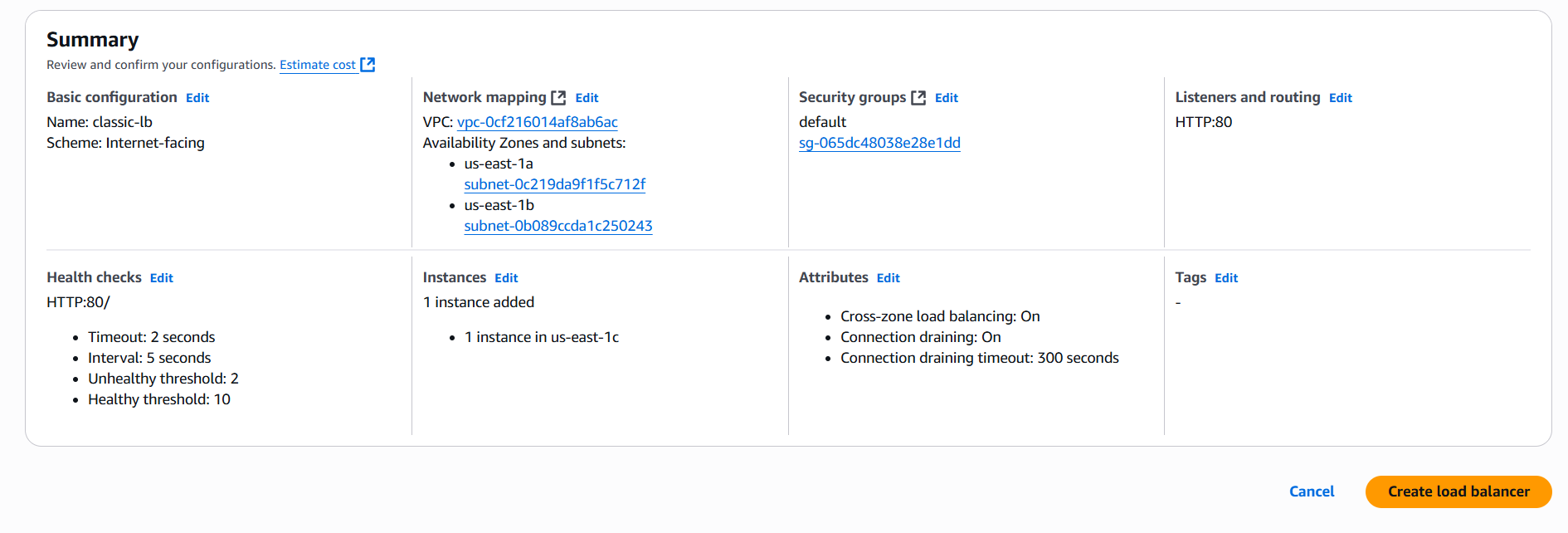
* Select the EC2 instances you want to balance traffic to.

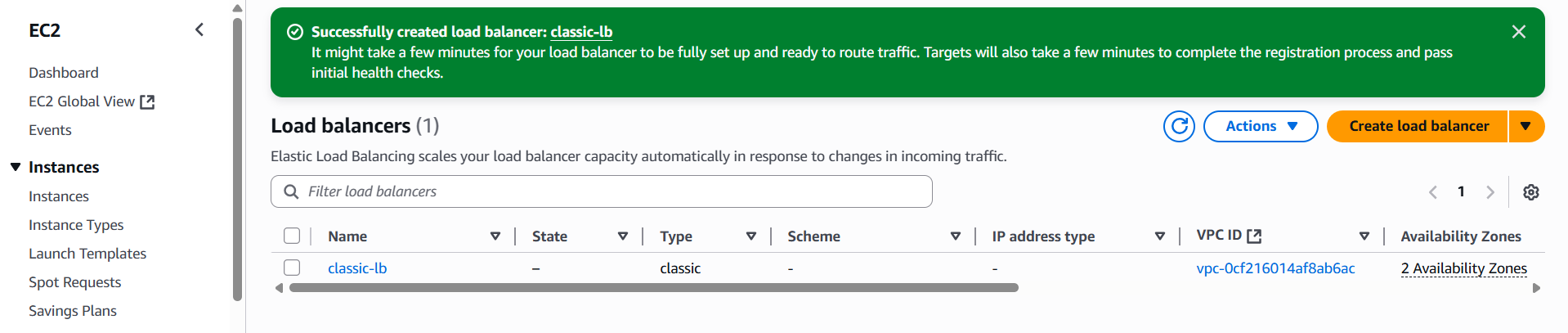
Click **Next**.

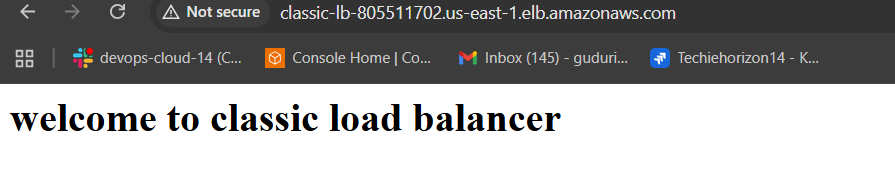


**6. Review & Create**

* Review all settings.
* Click **“Create”**.



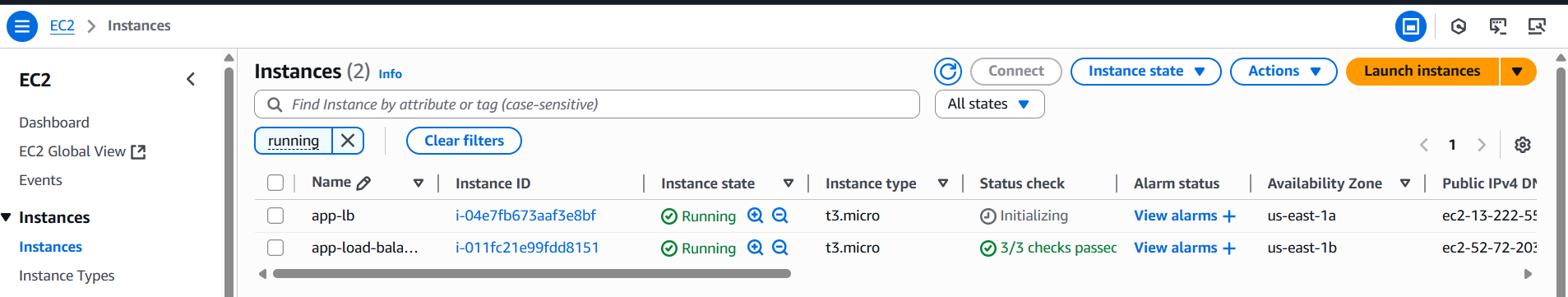


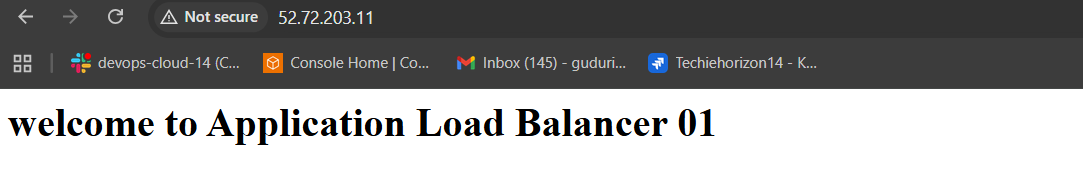


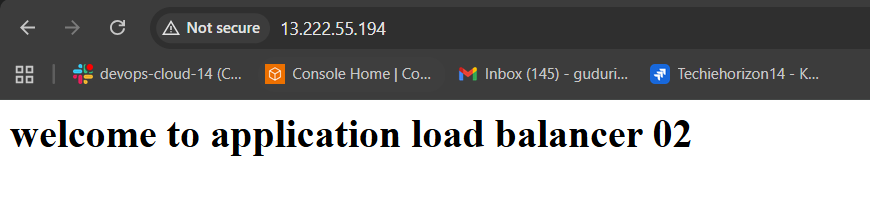
2) Configure Application Load balancer.

Create two instances with different availability zone

Follow the same steps as above to create the EC2 instances and install httpd on both

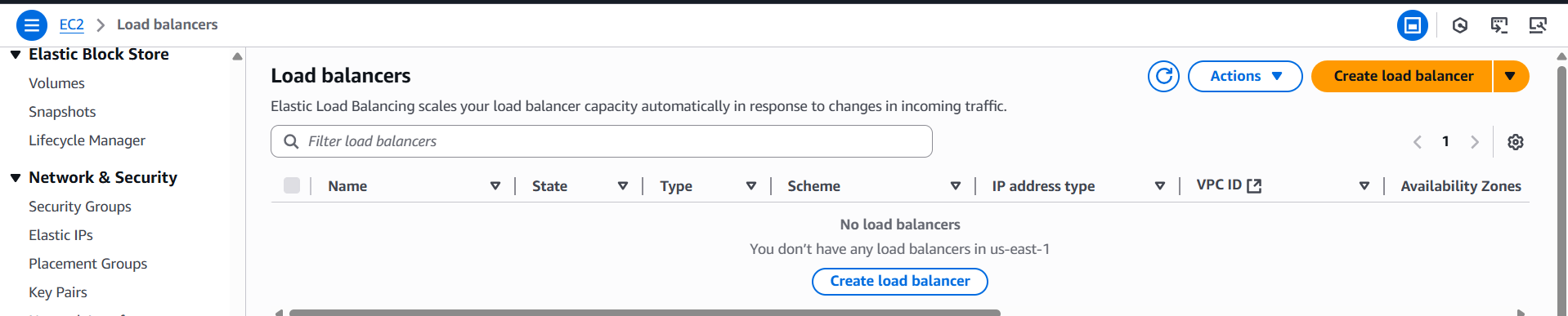






**1. Go to EC2 Dashboard**

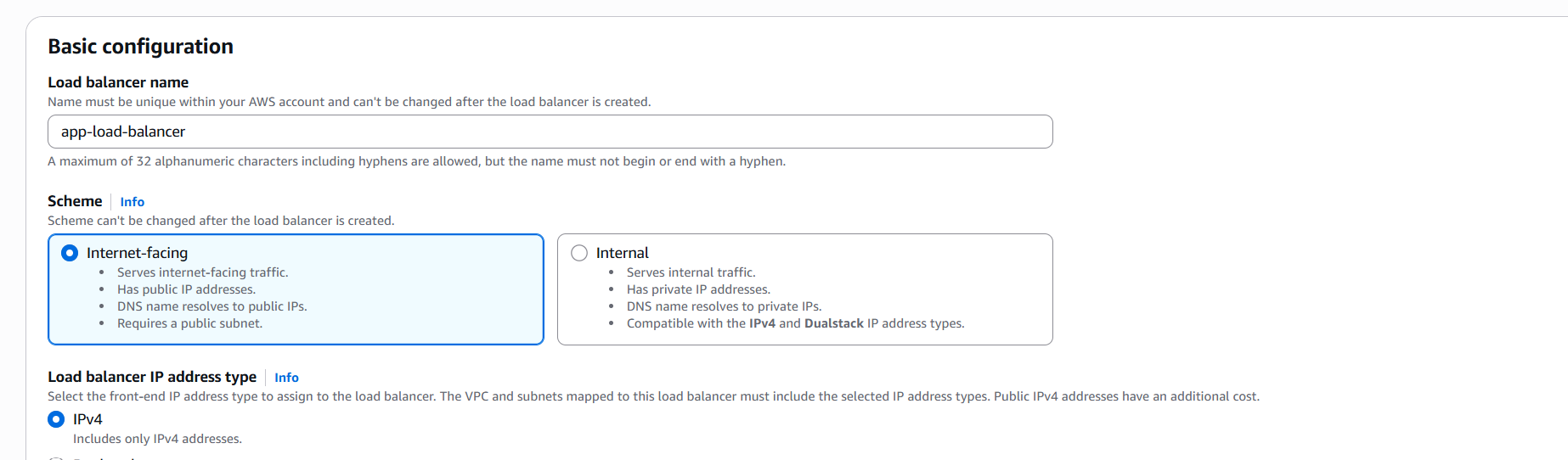
* Open the **AWS Console**.
* Navigate to **EC2 > Load Balancers**.
* Click **“Create Load Balancer”**.
* Choose **Application Load Balancer**.
* Click **“Create”**.



**2. Basic Configuration**

* **Name**: e.g., my-alb
* **Scheme**: Internet-facing (for public access) or Internal (for private access).
* **IP address type**: IPv4 (or dualstack if needed).

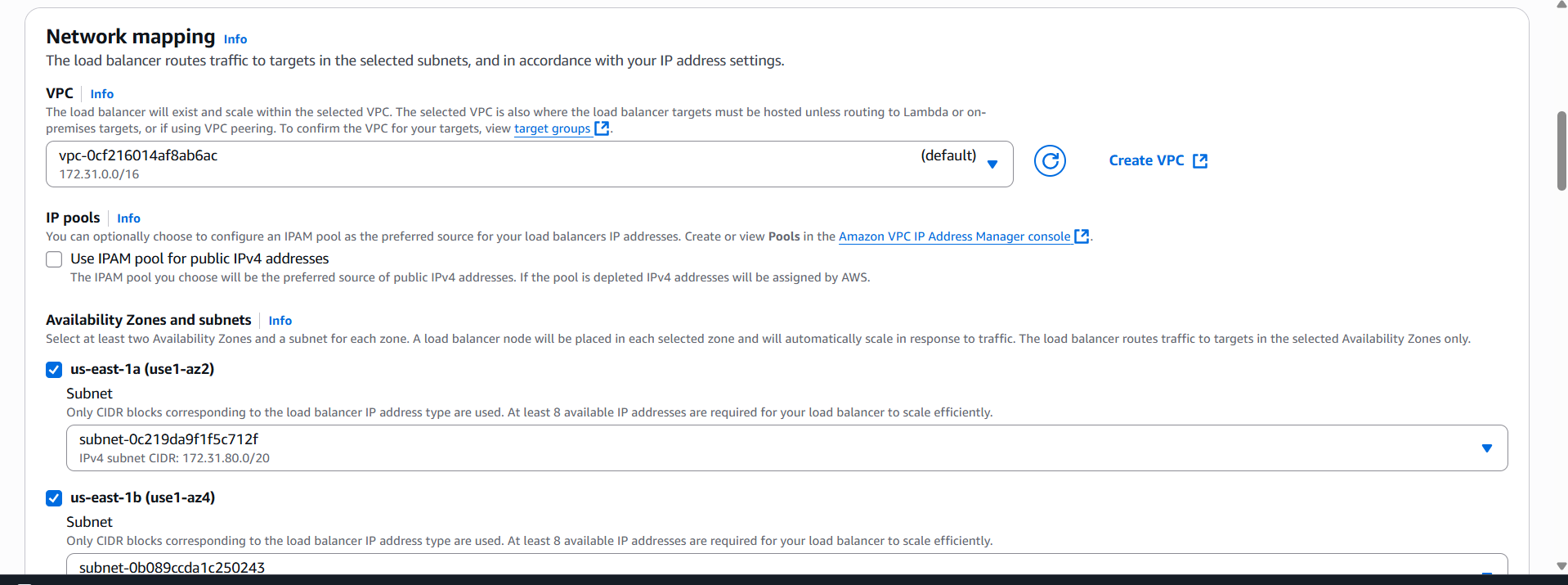
Click **Next**.



**3. Network Mapping**

* **VPC**: Select your VPC.
* **Availability Zones**: Select **at least 2 subnets** in different AZs.

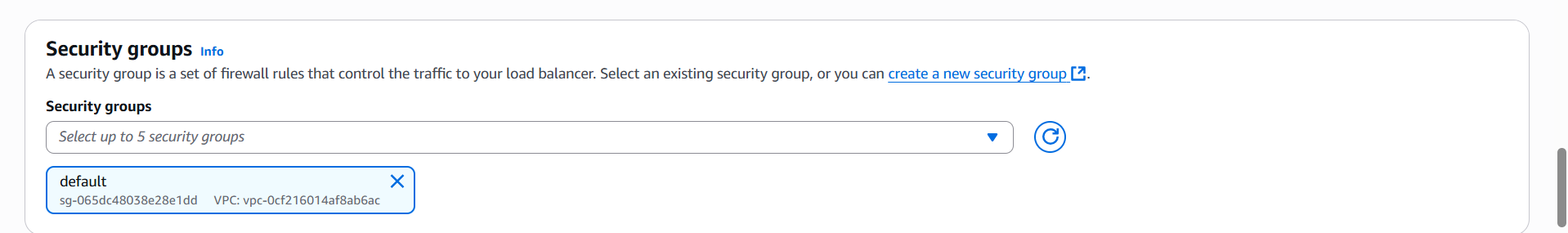
Click **Next**.



**4. Configure Security Groups**

* Select or create a **security group**.
* Make sure it allows:
  + **Inbound HTTP (port 80)** or HTTPS (port 443).
  + Other ports if needed.

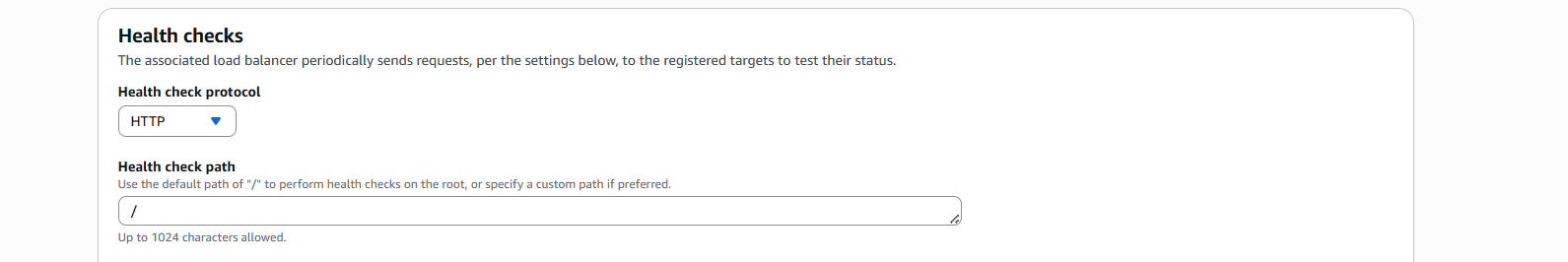
Click **Next**.

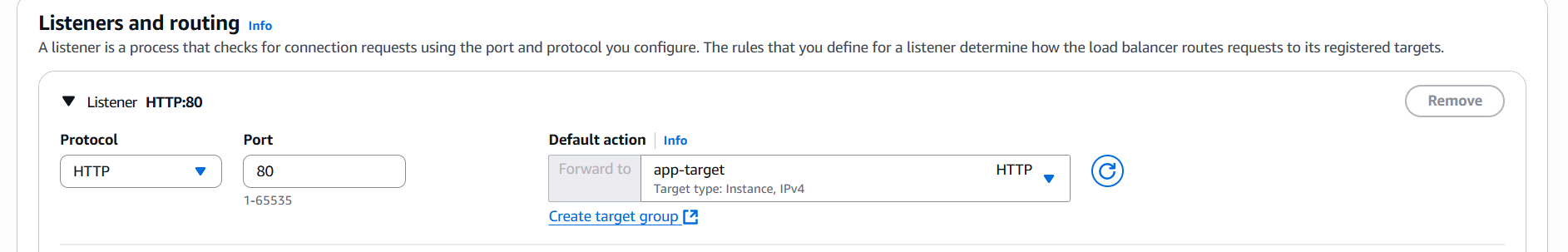


**5. Configure Listeners and Routing**

* **Listener**: Default is HTTP on port 80.
* **Target Group**:
  + Name: my-target-group
  + Target type: Instance (or IP/ Lambda)
  + Protocol: HTTP
  + Port: 80
  + Health check path: / or a custom path.

Click **Next**.

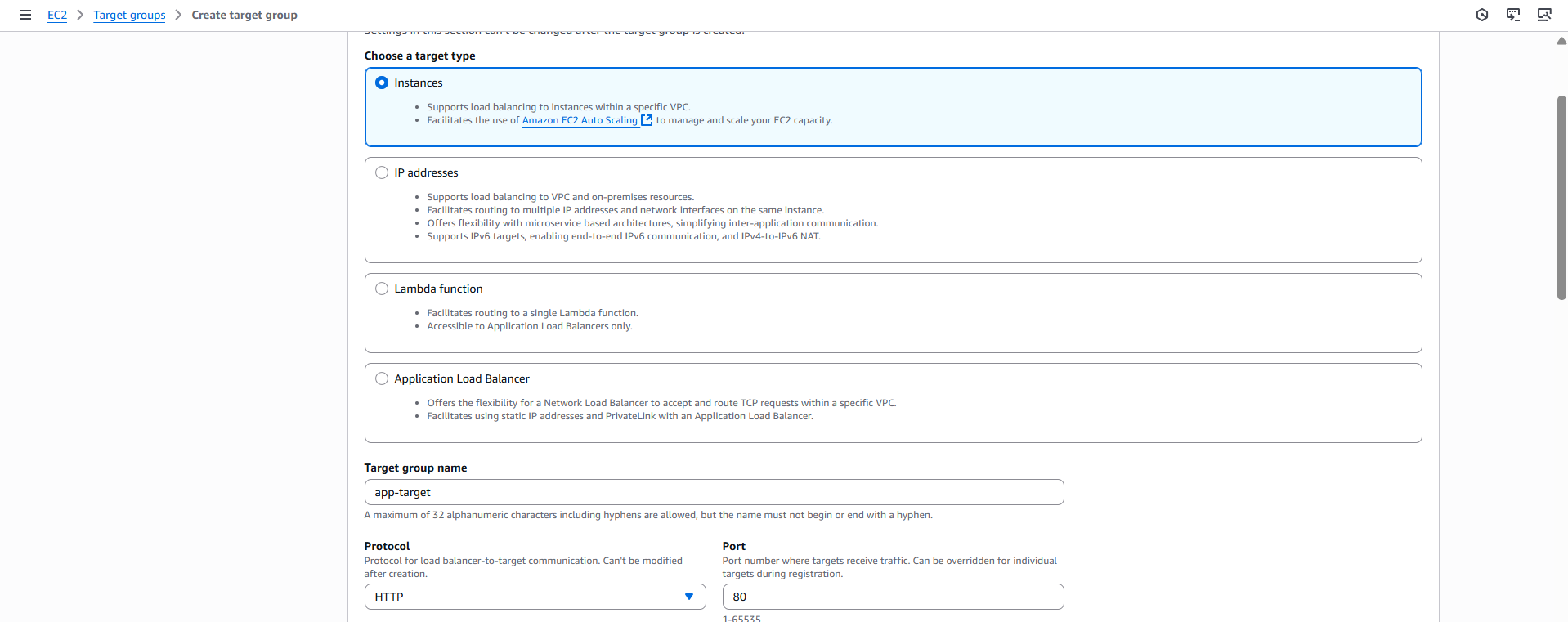


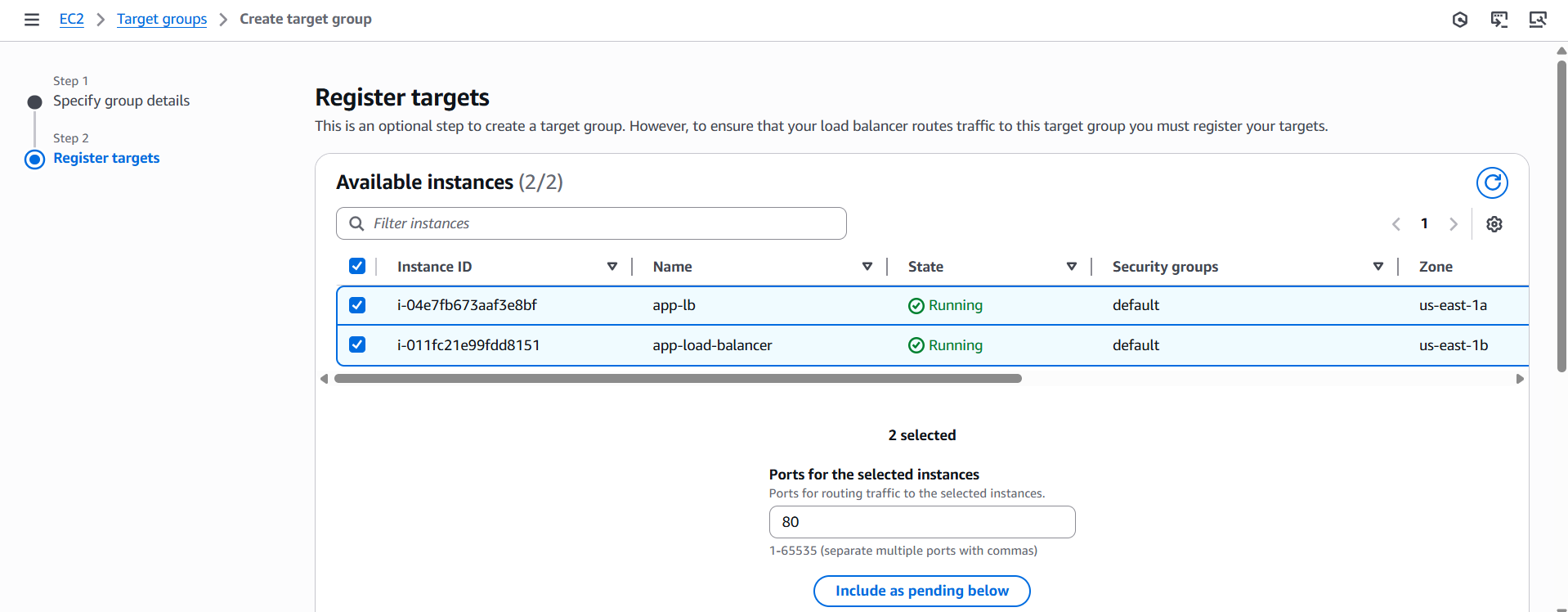


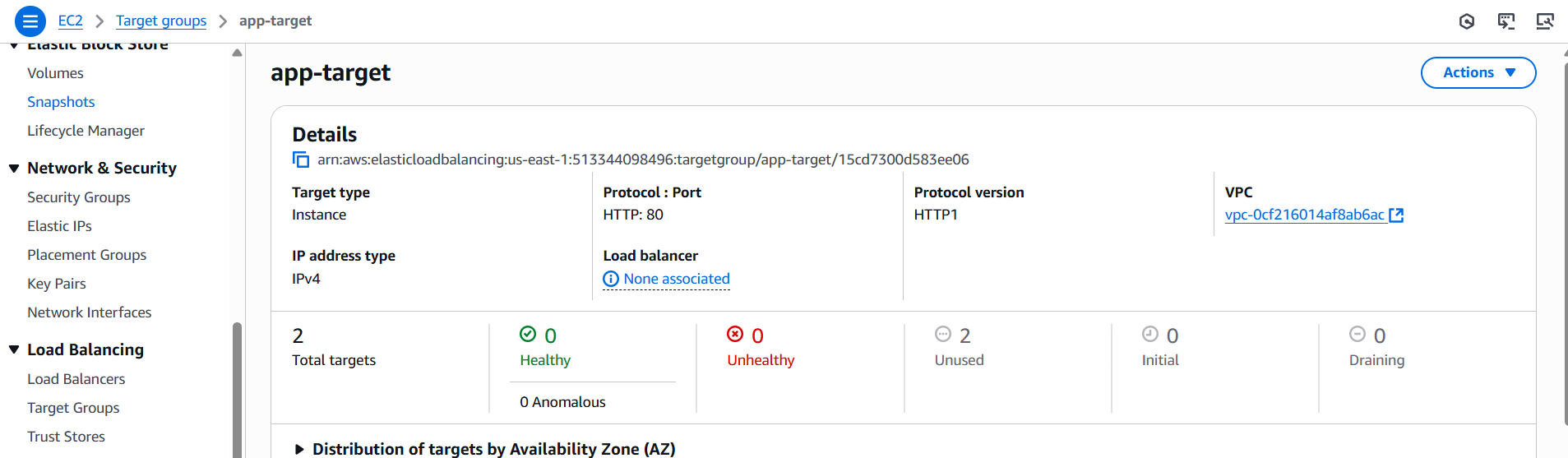
**6. Register Targets**

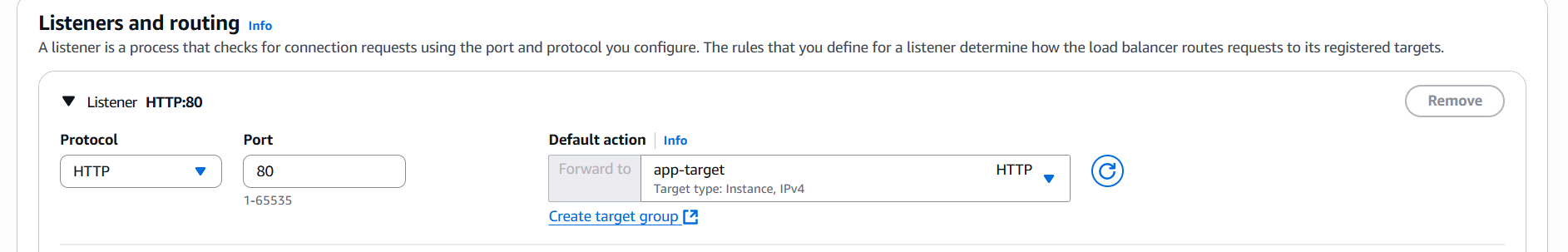
* Select EC2 instances to add to the target group.
* Click **Include as pending below**.

Click **Next**.



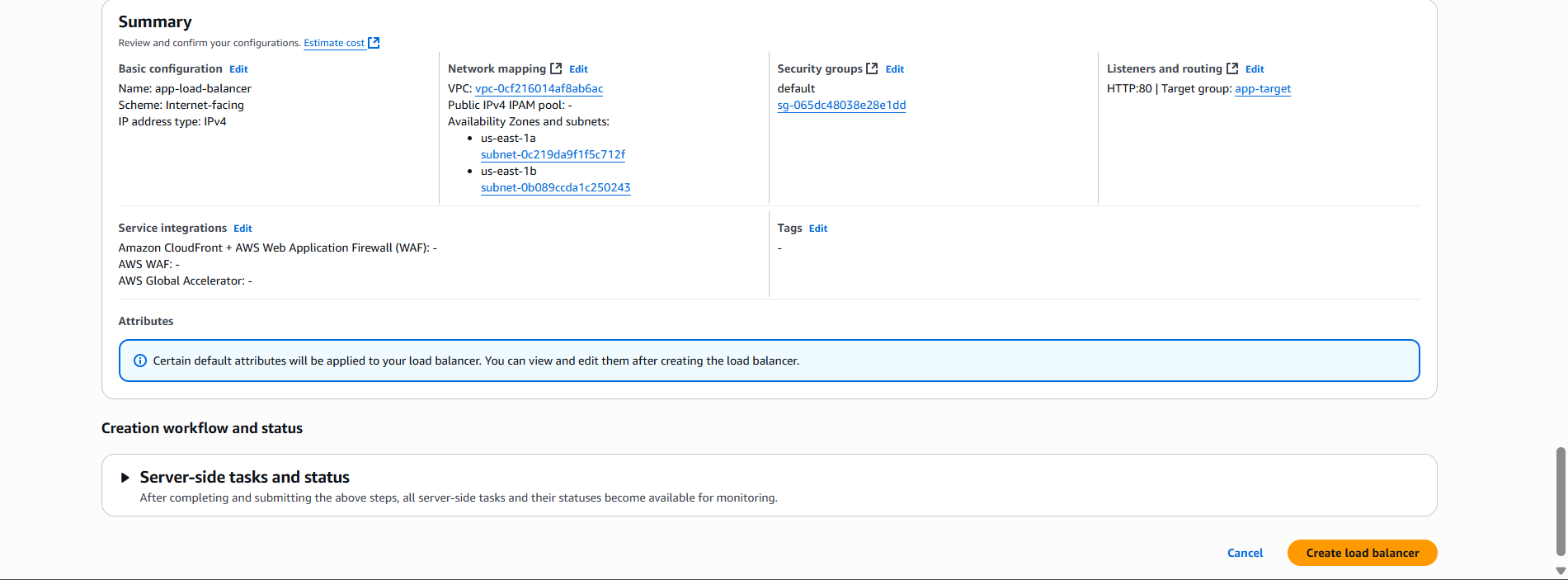






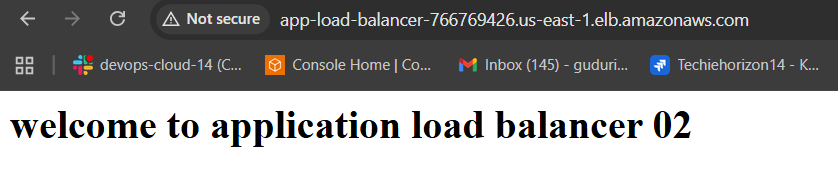
**7. Review and Create**

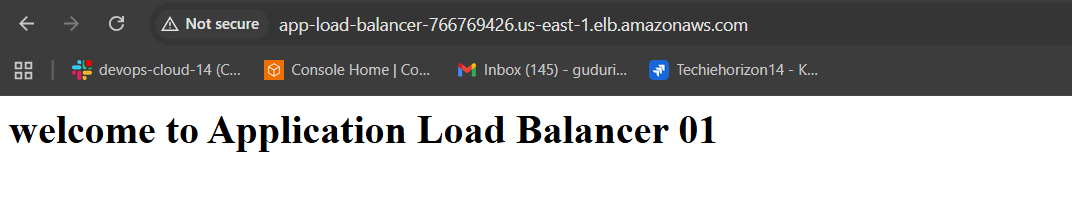
* Review all settings.
* Click **Create Load Balancer**.



Use the DNS name:

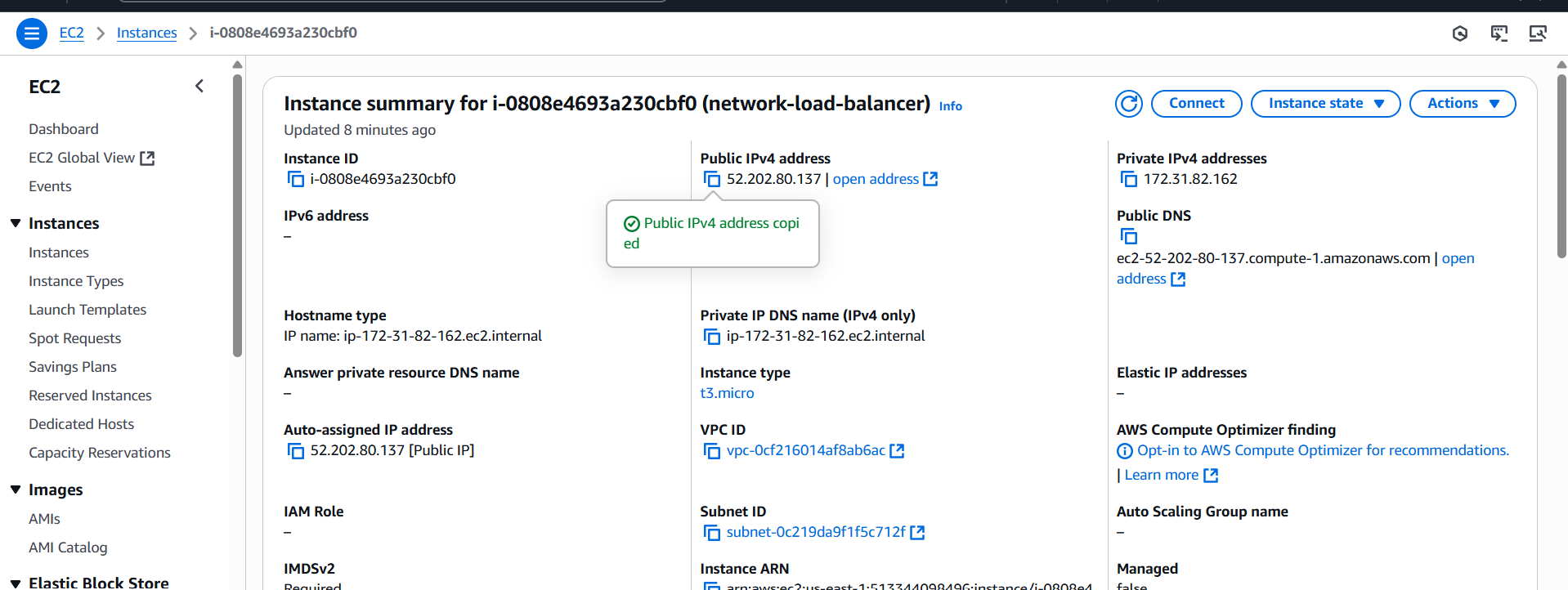
app-load-balancer-766769426.us-east-1.elb.amazonaws.com

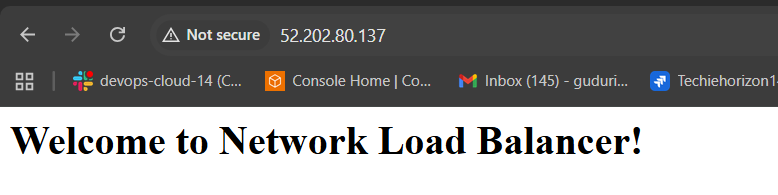




3) Configure Network Load balancer.

Create an EC2 Instance using above steps and install nginx in it.





Create Elastic IP address:

1. Log in to the **AWS Management Console**.

2. Go to **EC2** service.

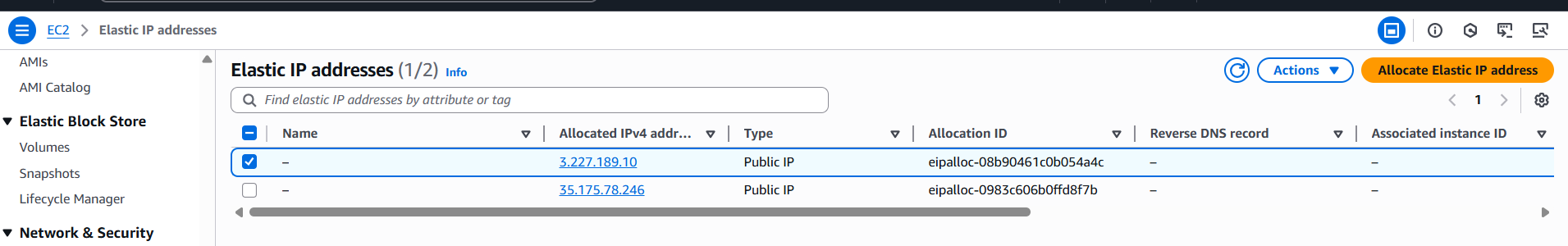
3. In the left sidebar, scroll down and click **“Elastic IPs”** under **Network & Security**.

4. Click the **“Allocate Elastic IP address”** button.

5. Choose the **Amazon’s pool of IPv4 addresses** (default).

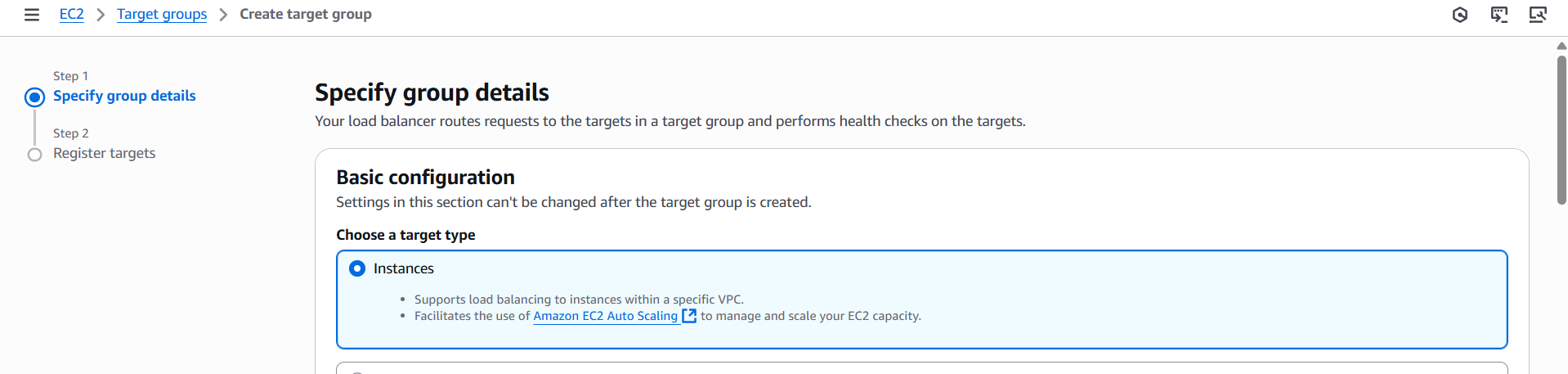
6. Click **Allocate**.

7. You’ll get a new Elastic IP address ready to assign to an instance, load balancer, or network interface.

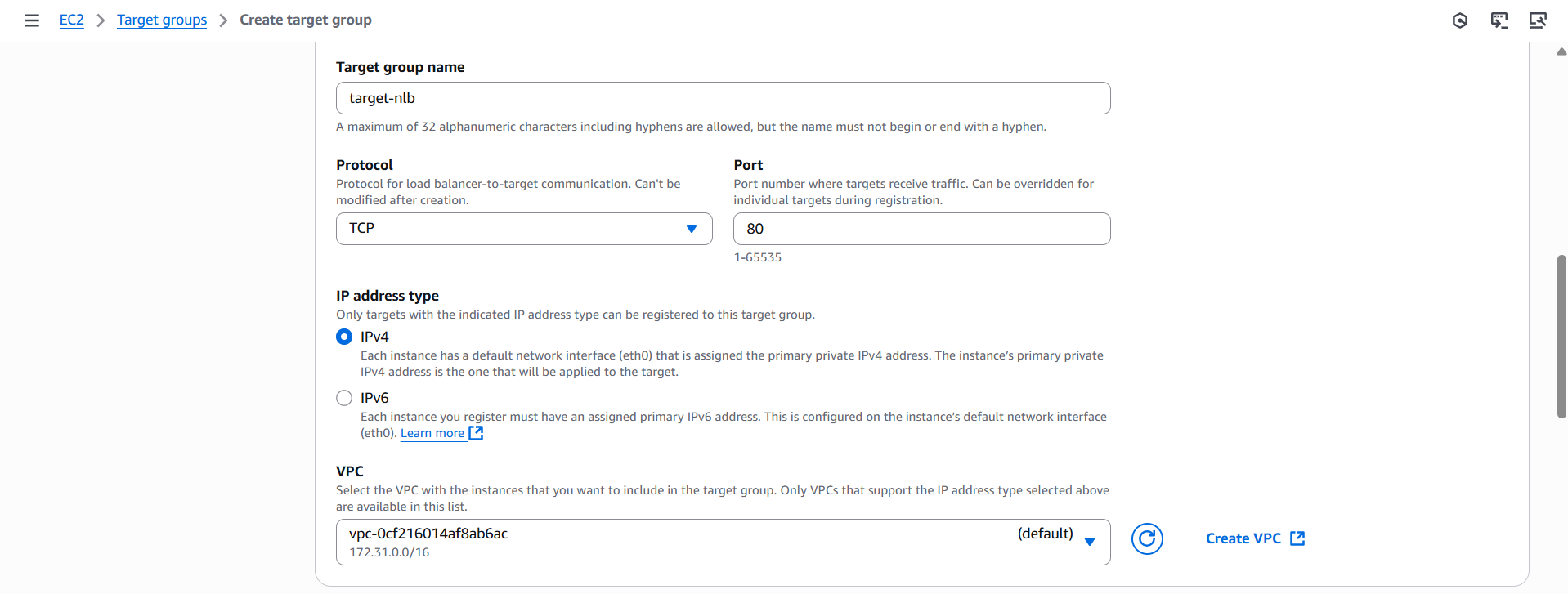


Create a target group for NLB:

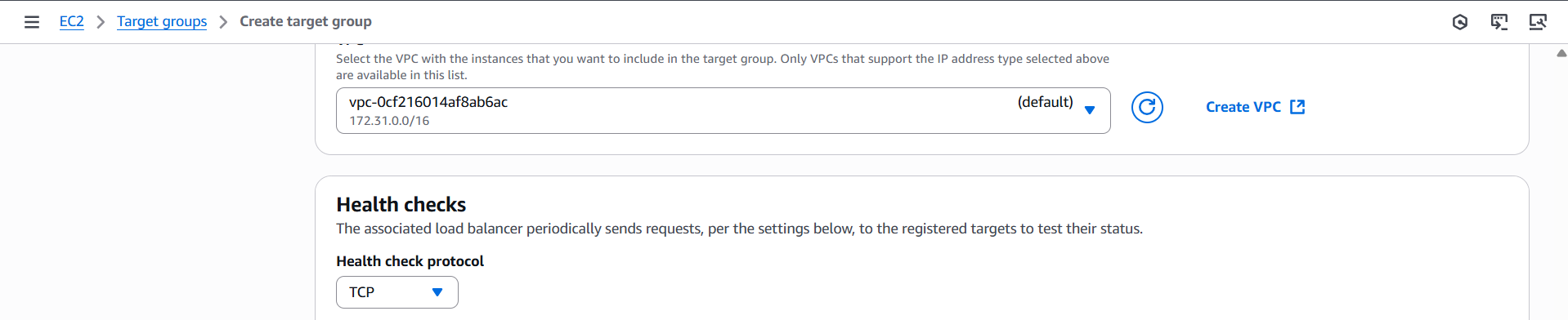
1. **Go to the EC2 Dashboard**:
   * Open AWS Console → EC2 → Load Balancing → **Target Groups**.
2. **Click "Create target group"**.
3. **Choose target type**:
   * For NLB, valid types are:
     + Instance (for EC2 instances)



1. **Set these values**:
   * **Target group name**: e.g., target-nlb
   * **Protocol**: TCP, TLS, or UDP (depending on your app)
   * **Port**: The port your targets will receive traffic on (e.g., 80, 443)
   * **VPC**: Choose the VPC where your targets are located.



1. **Health checks**:
   * Protocol: TCP (recommended for NLB)



1. Register Targets:

Select the instance which created with nginx installed.

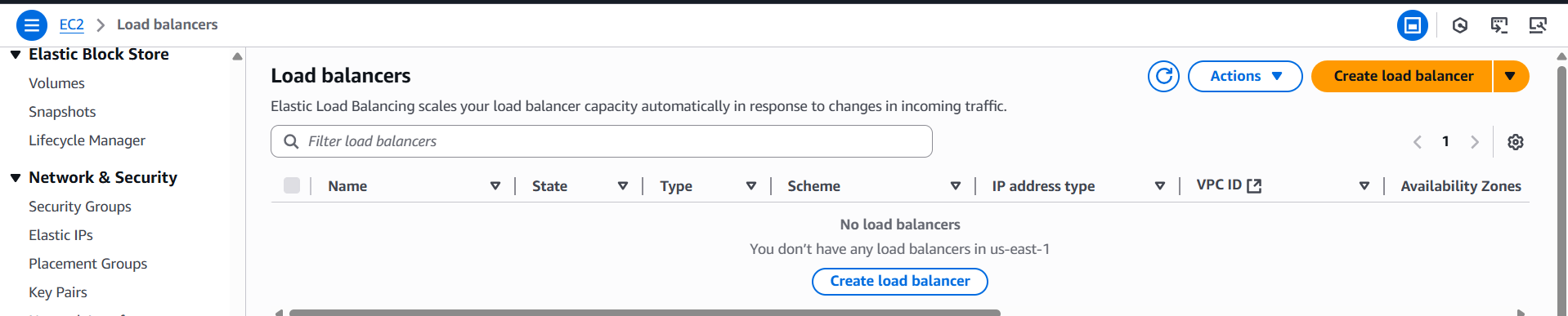


7. Click create target.

Configure the Network Load Balancer.

**1. Go to EC2 Dashboard**

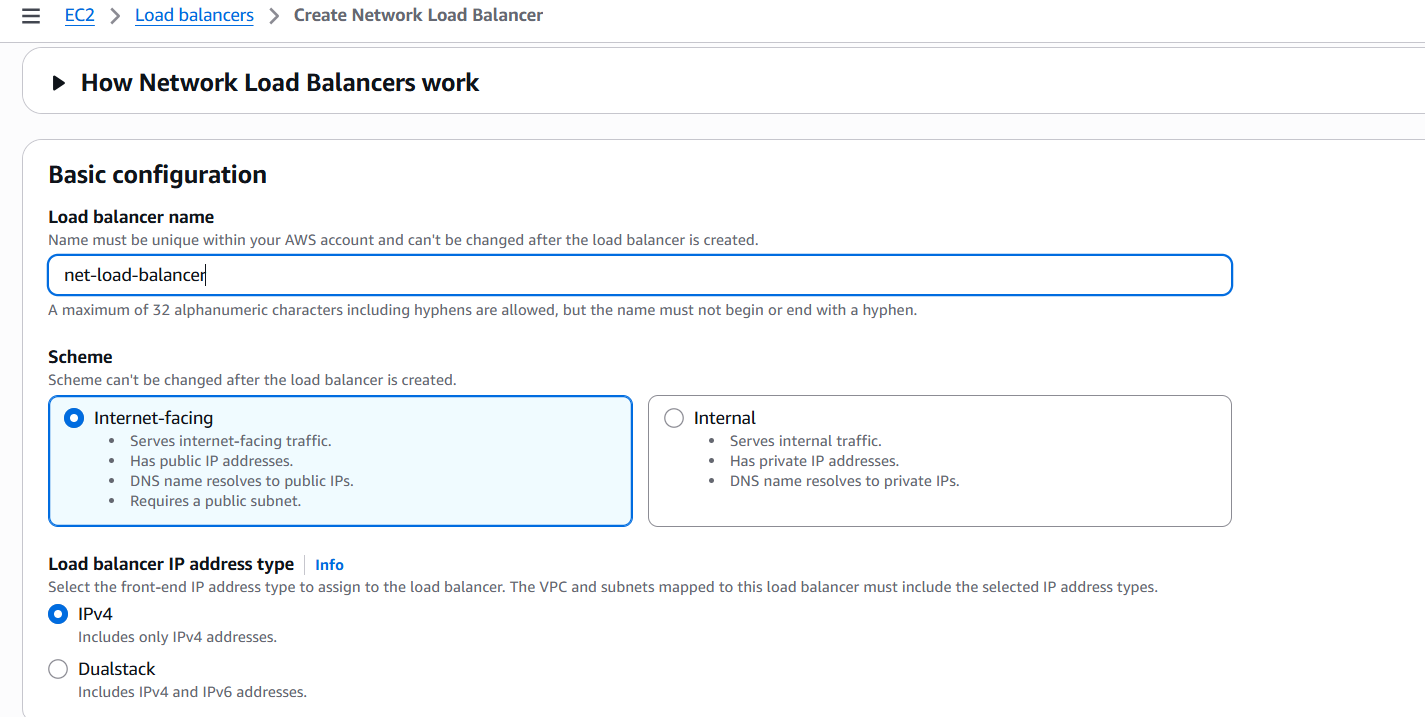
* Open the **AWS Management Console**.
* Go to **EC2 > Load Balancers**.
* Click **“Create Load Balancer”**.
* Choose **Network Load Balancer**.
* Click **“Create”**.



**2. Basic Configuration**

* **Name**: e.g., net-load balancer
* **Scheme**:
  + Internet-facing (for public access)
* **IP address type**: IPv4

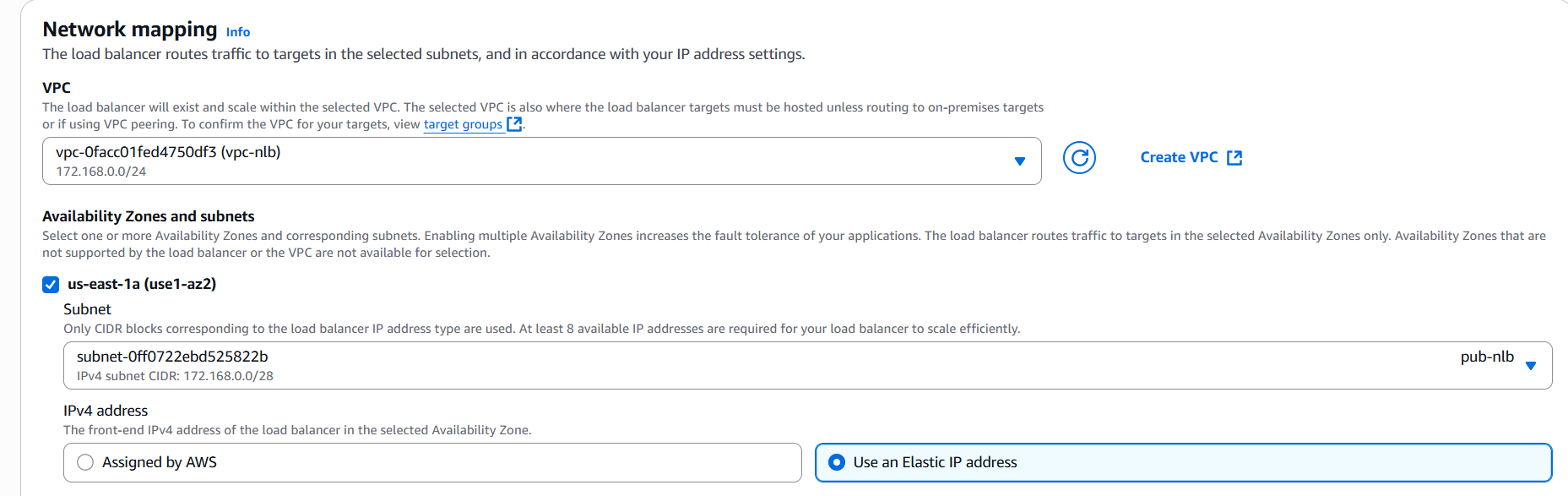
Click **Next**.

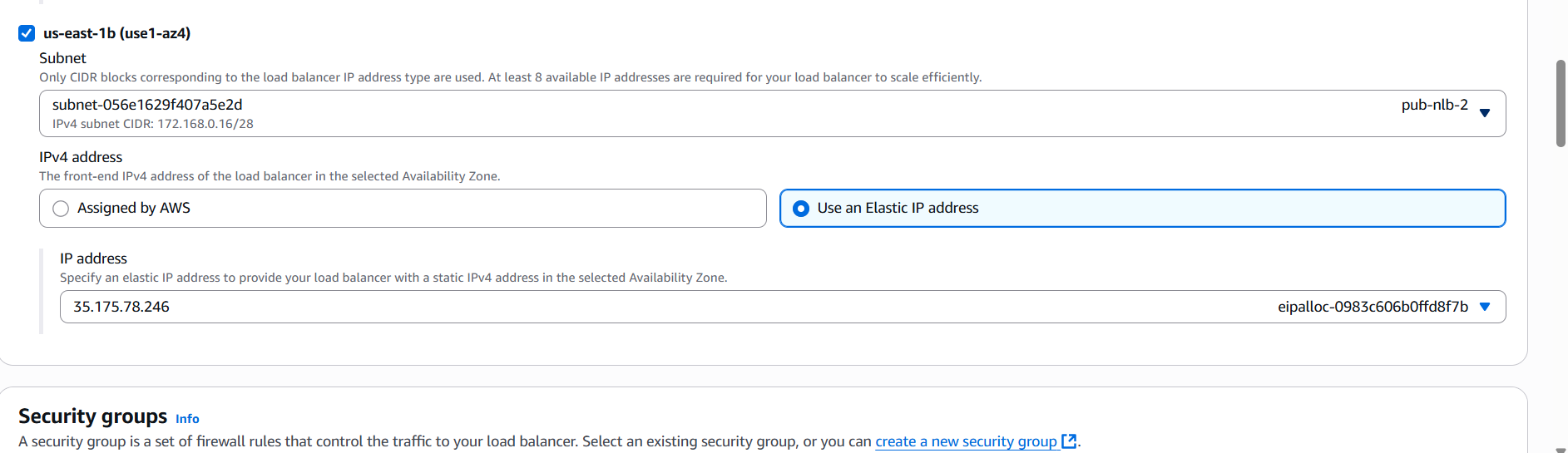


**3. Network Mapping**

* **VPC**: Select your VPC.
* **Availability Zones**: Select **at least 2 subnets**, one per AZ.
* You can assign **Elastic IPs** (optional, for static IP).

Click **Next**.

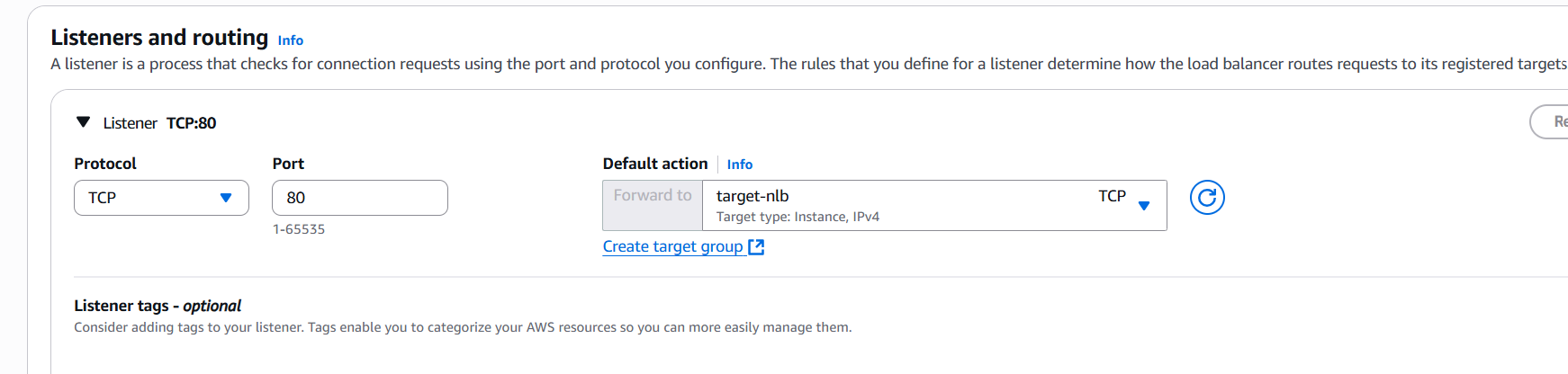




**4. Listeners and Routing**

* **Listener protocol**: TCP
* **Port**: 80 (or your app’s port)
* **Target group**:
  + Name: my-nlb-targets
  + Target type: Instance or IP
  + Protocol: TCP
  + Port: 80 (or match your app)

Click **Next**.



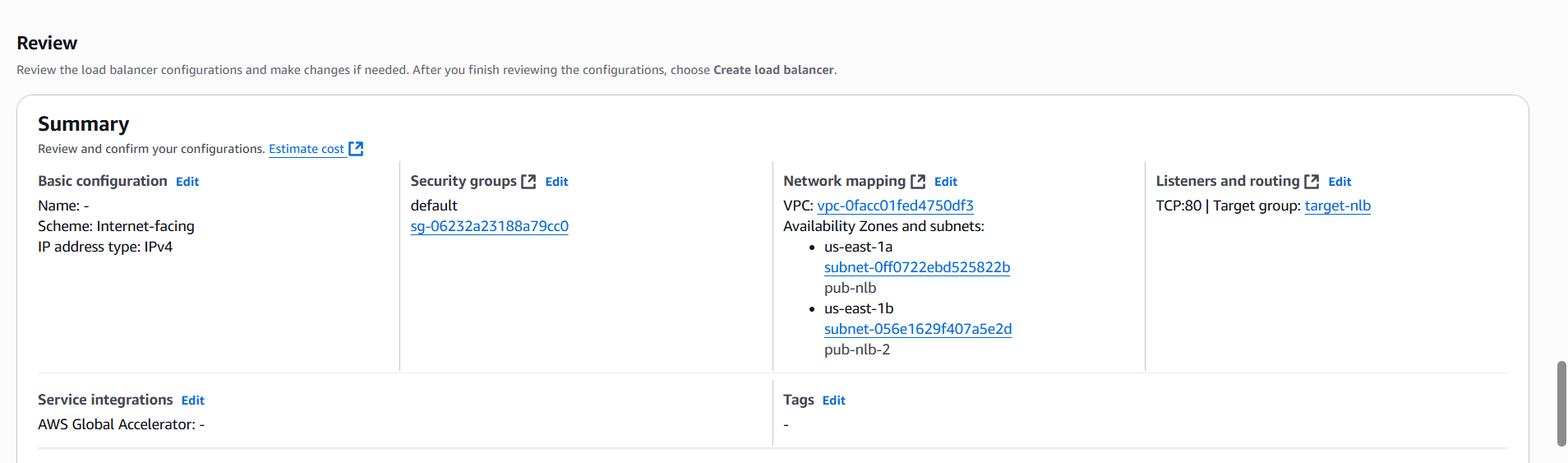
**5. Register Targets**

* Select EC2 instances to handle traffic.
* Click **Include as pending below**.

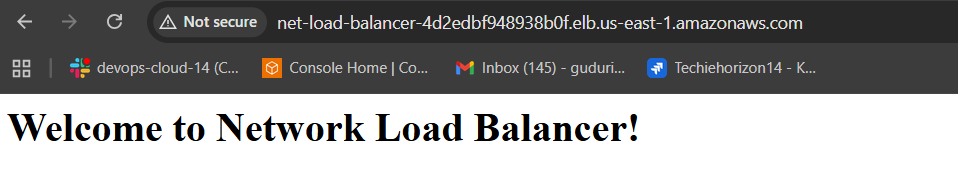
Click **Next**.

**6. Review and Create**

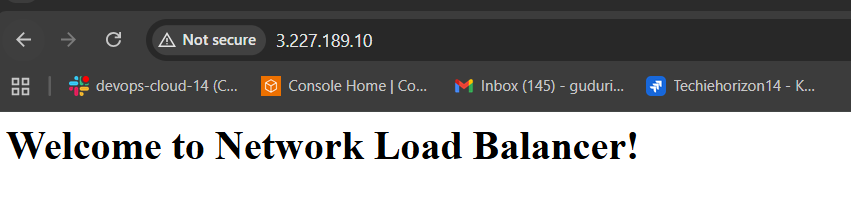
* Review all settings.
* Click **“Create Load Balancer”**.



Use DNS name: http://net-load-balancer-4d2edbf948938b0f.elb.us-east-1.amazonaws.com/



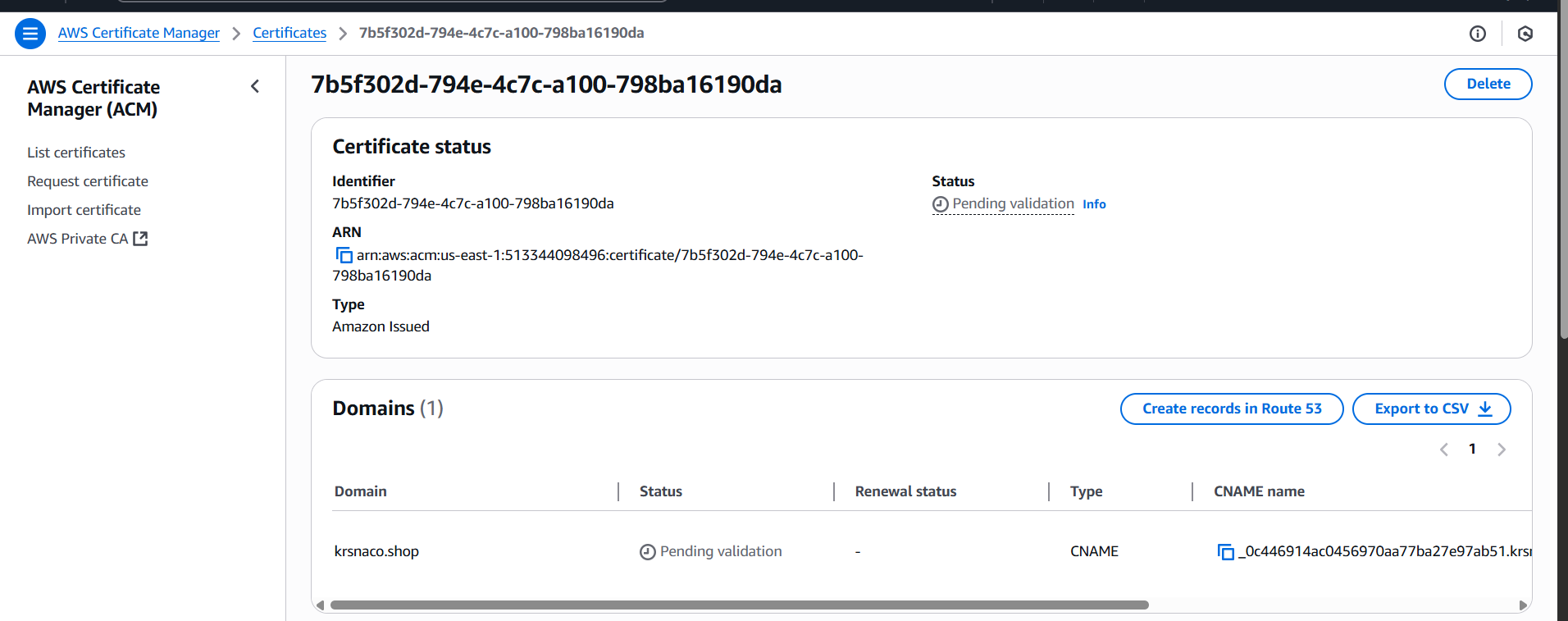
Using Elastic IP: http://3.227.189.10/

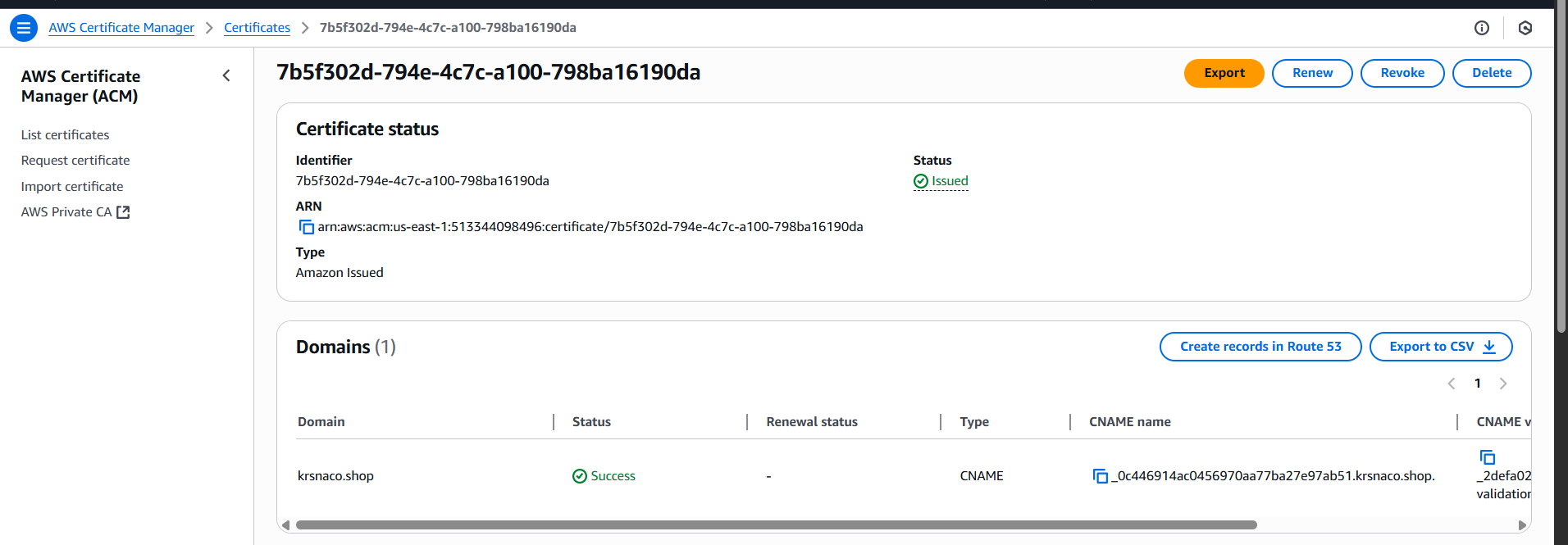


4) Attach SSL for application load balancer.

**1. Get or Request a Certificate in ACM**

* Go to **AWS Certificate Manager (ACM)**.
* Request a **public certificate** for your domain (e.g., example.com, \*.example.com).
* Validate it using **DNS** or **Email**.
* Once issued, you’ll have the ARN of the certificate.

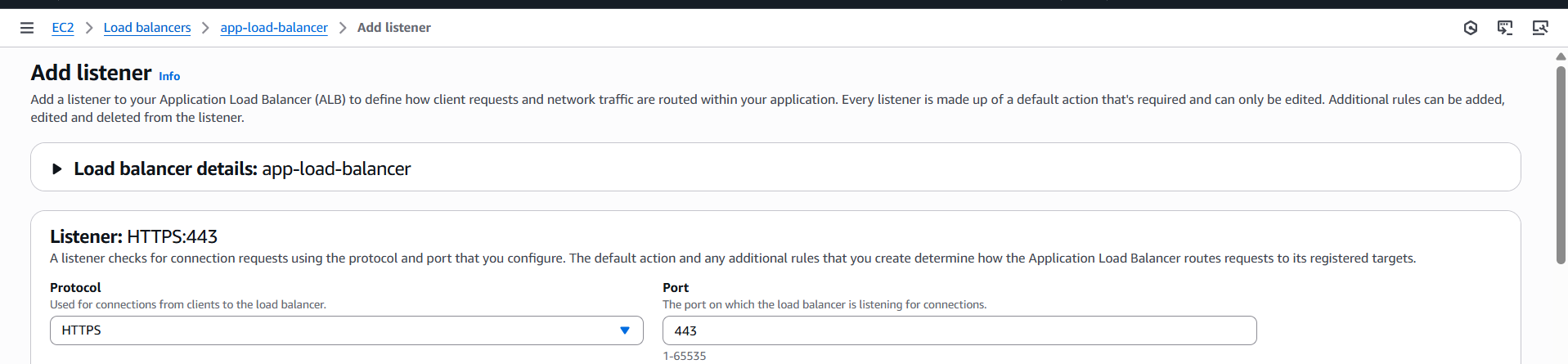


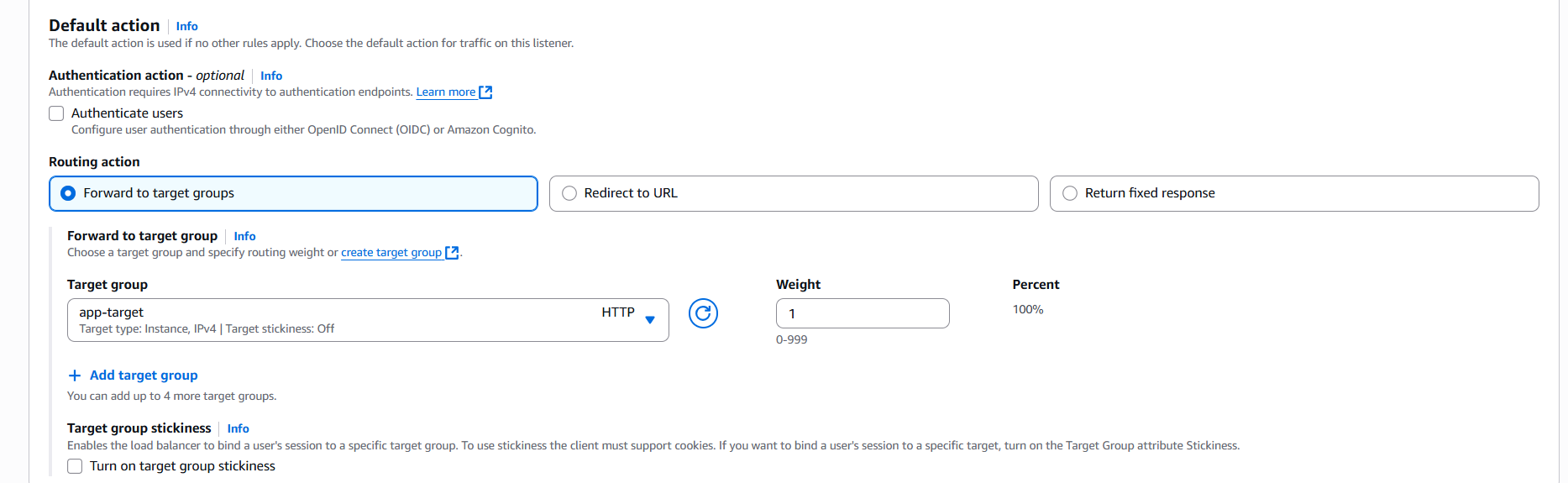


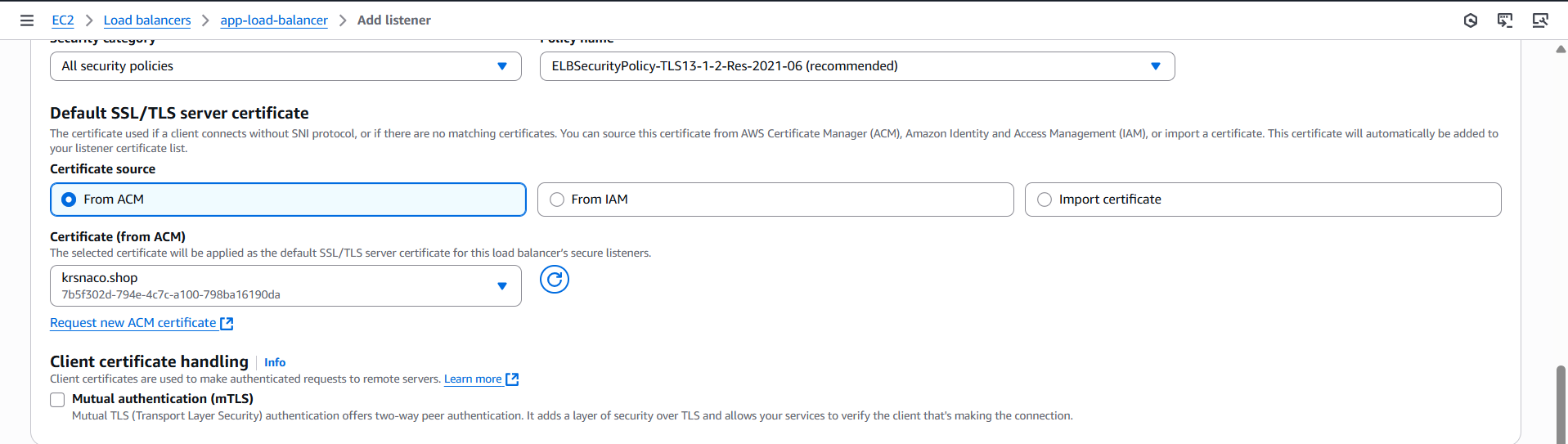
**2. Add HTTPS Listener to ALB with the SSL Certificate**

**Via AWS Console:**

1. Open **EC2 Console** > **Load Balancers**.
2. Select your **Application Load Balancer**.
3. Go to the **Listeners** tab.
4. Click **Add listener**.
5. Choose **HTTPS** as the protocol.
6. Set port **443**.
7. Select the **ACM certificate** from the dropdown.
8. Set the default action (forward to your target group).
9. Save.









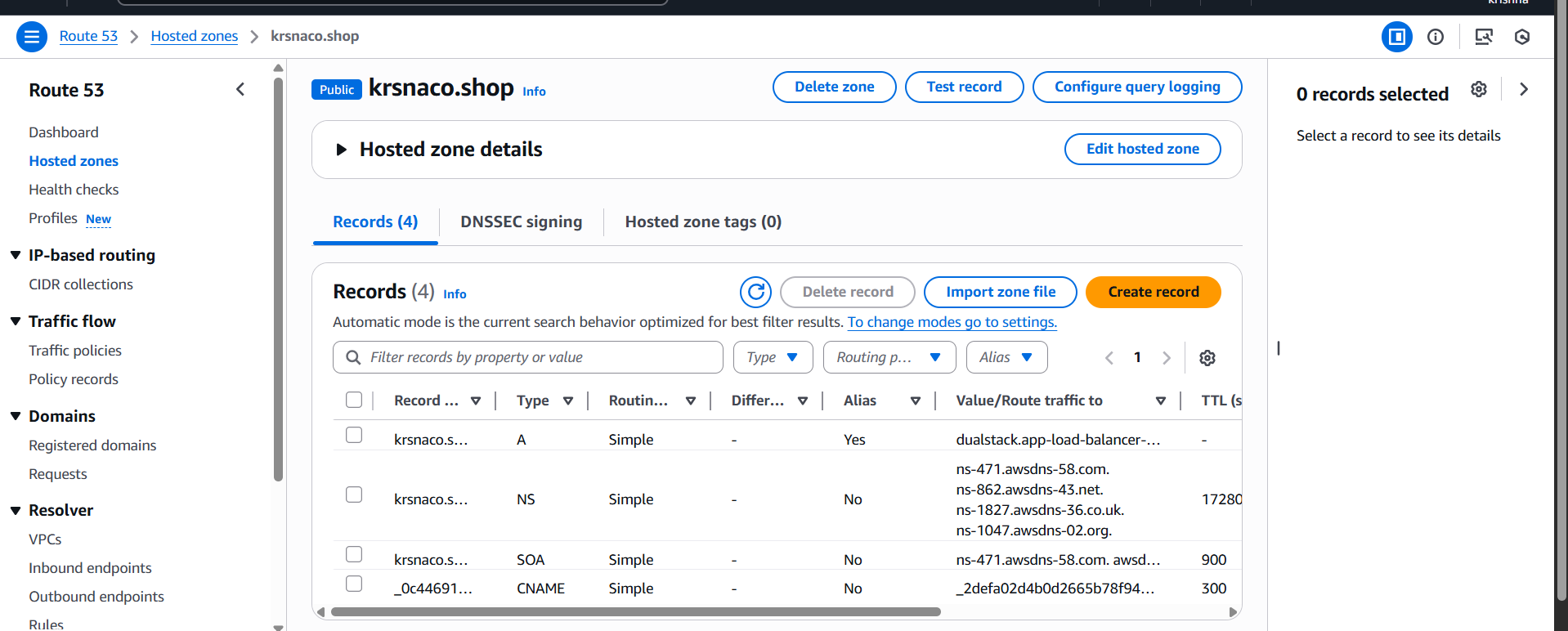
5) Map Application load balancer to R53.

**Step 1: Go to Route 53**

1. Open the **Route 53 Console**.
2. Navigate to **Hosted zones**.
3. Select your hosted zone for the domain (e.g., krsnaco.shop).

**Step 2: Create Record**

1. Click **“Create record”**.
2. Enter the **subdomain or root domain**:
   * leave blank for root domain (example.com)
3. Choose **Record type**: A – IPv4 address
4. Choose **Alias**: Yes
5. **Alias target**: Choose your **Application Load Balancer** from the dropdown list.
   * AWS automatically recognizes ALBs in the same region.
6. Click **Create records**.



6) Push the application load balancer logs to s3

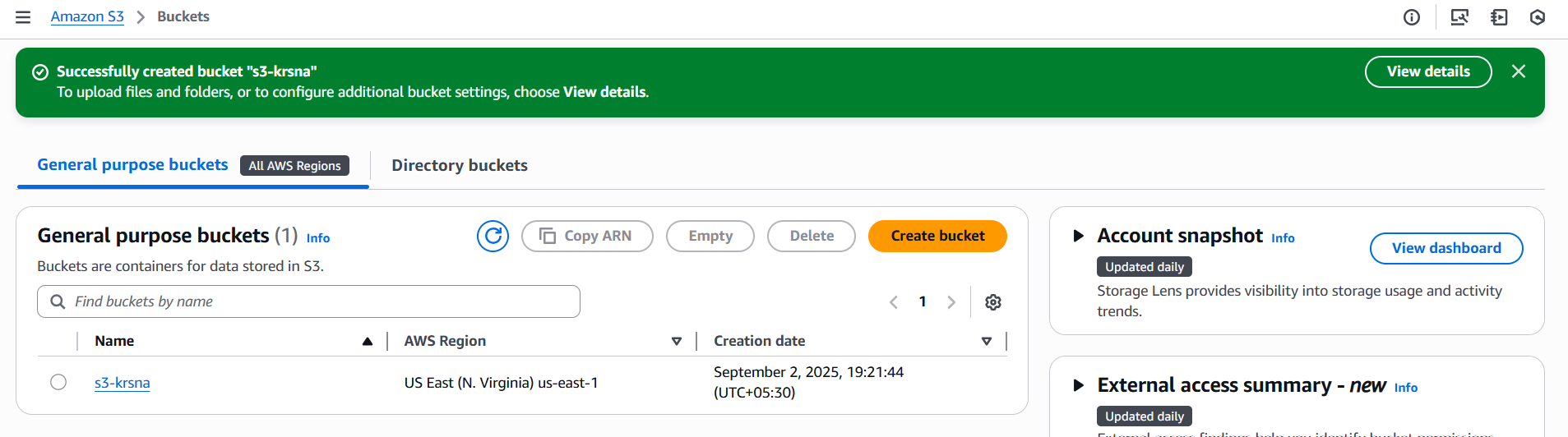
**1. Create or choose an S3 Bucket**

You need a bucket in the **same region** as the ALB.

**Create a new bucket**

Go to S3 → Create bucket  
Example: s3-krsna

Make sure to **unblock public access** settings *only if needed* (logs don't need to be public).



**2. Attach a Bucket Policy for ALB Logging**

ALB needs permission to write logs into the bucket.

{

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

"Statement": [

{

"Sid": "AWSALBLoggingPermissions",

"Effect": "Allow",

"Principal": {

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

},

"Action": "s3:PutObject",

"Resource": "arn:aws:s3:::s3-krsna/AWSLogs/513344098496/\*",

"Condition": {

"StringEquals": {

"aws:SourceAccount": "513344098496"

},

"ArnLike": {

"aws:SourceArn": "arn:aws:elasticloadbalancing:us-east-1:513344098496:loadbalancer/app/\*"

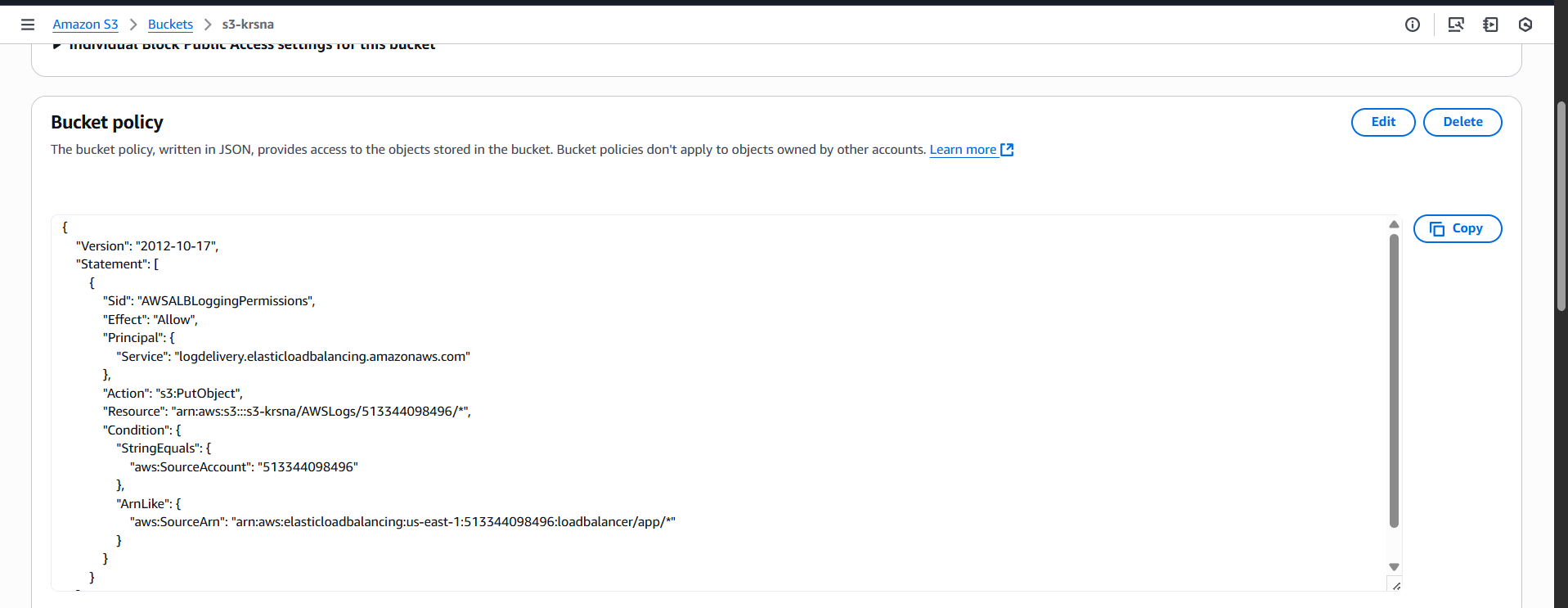
}

}

}

]

}



**3. Enable Access Logging on the ALB**

1. Go to **EC2 Console** > **Load Balancers**.
2. Select your **Application Load Balancer**.
3. Go to the **Description** tab.
4. Click **Edit attributes**.
5. Under **Access logs**, check **Enable**.
6. Fill in:
   * **S3 location**: s3-krsna
   * **Prefix** (optional): e.g., alb-logs/
7. Save.

