1. Configure Classic Load balancer.

**Access the EC2 Dashboard**:

* Log in to the AWS Management Console.
* Navigate to **Services** > **EC2**.
* In the left-hand menu, under **Load Balancing**, select **Load Balancers**. **Create Load Balancer**:
* Click on **Create Load Balancer**.
* Choose **Classic Load Balancer** and click **Create**.

**Configure Load Balancer**:

* **Name**: Assign a unique name to your load balancer.
* **Scheme**: Select **Internet-facing** for public access or **Internal** for private access within a VPC.
* **Listeners**: Define the protocol and port for front-end (client to load balancer) and back-end (load balancer to instance) connections. For example:
  + Front-end: HTTP, port 80
  + Back-end: HTTP, port 80
* **Availability Zones**: Select the VPC and the Availability Zones where your instances are located. **Assign Security Groups**:
* Choose an existing security group or create a new one that allows inbound traffic on the listener ports (e.g., 80, 443).

**Configure Health Check**:

* Set up health check parameters to monitor the health of your instances:
  + **Protocol**: HTTP
  + **Port**: 80
  + **Ping Path**: / (or a specific path like /healthcheck)
  + **Response Timeout**: 5 seconds
  + **Interval**: 30 seconds
  + **Unhealthy Threshold**: 2
  + **Healthy Threshold**: 10

Add **EC2 Instances**:

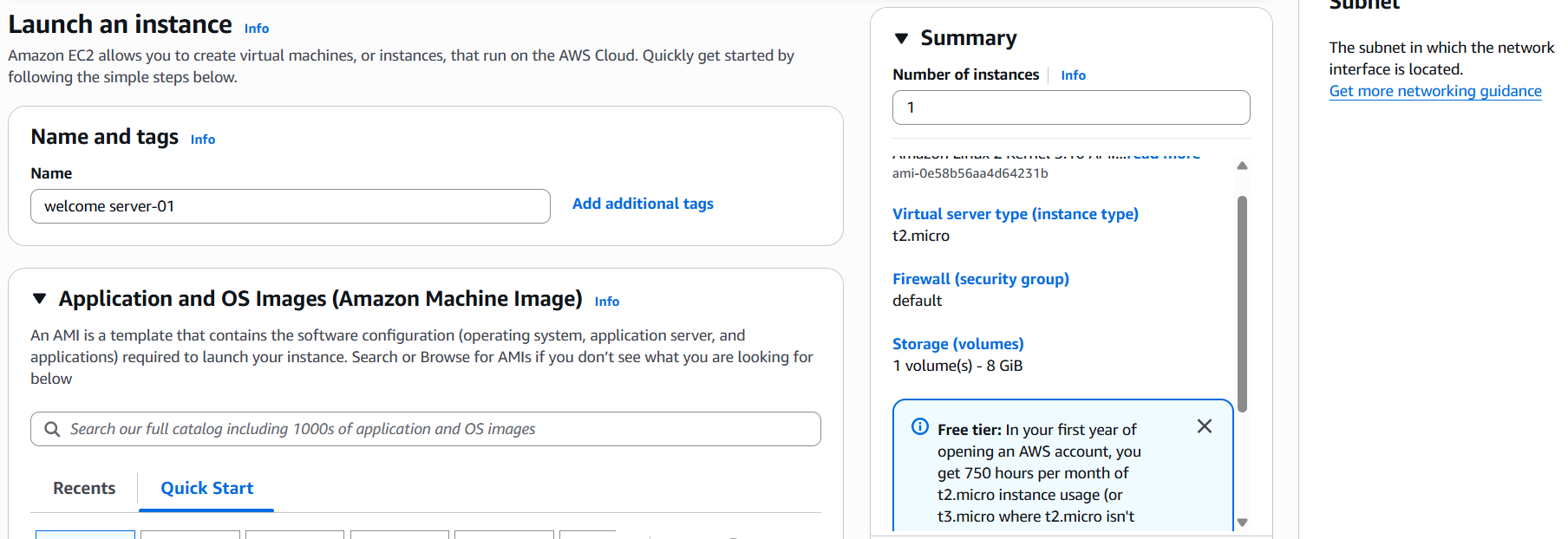
* Select the instances you want to register with the load balancer.
* Click **Add to Registered**.

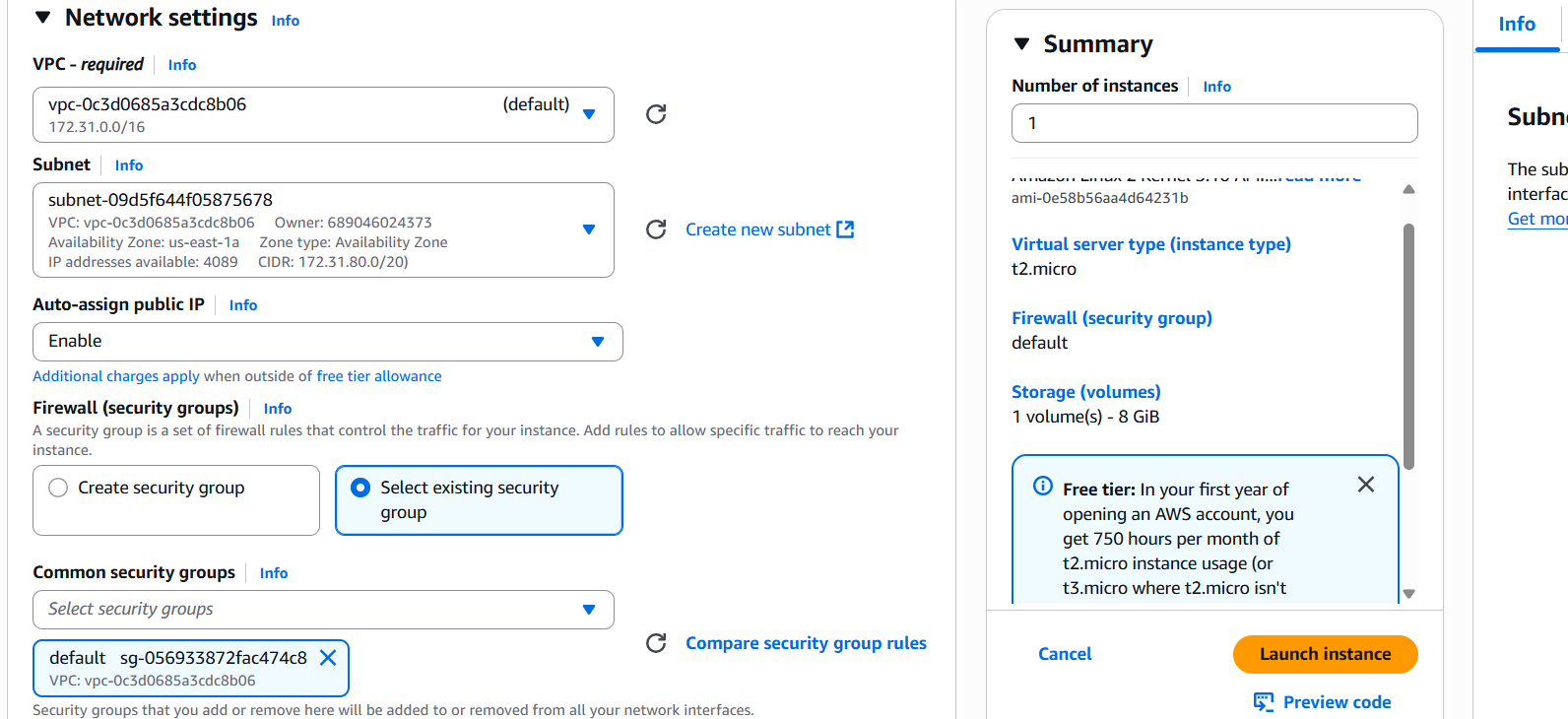
**Add Tags (Optional)**:

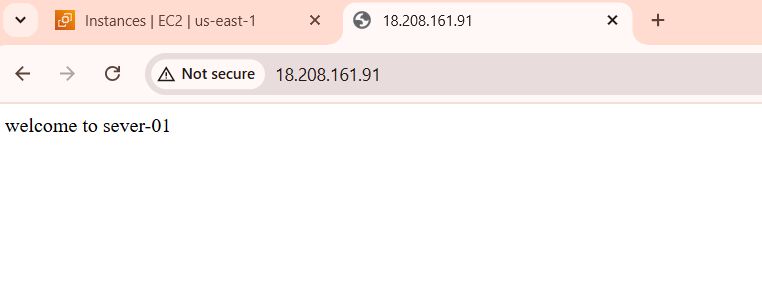
* Assign tags to organize and manage your load balancer resources.

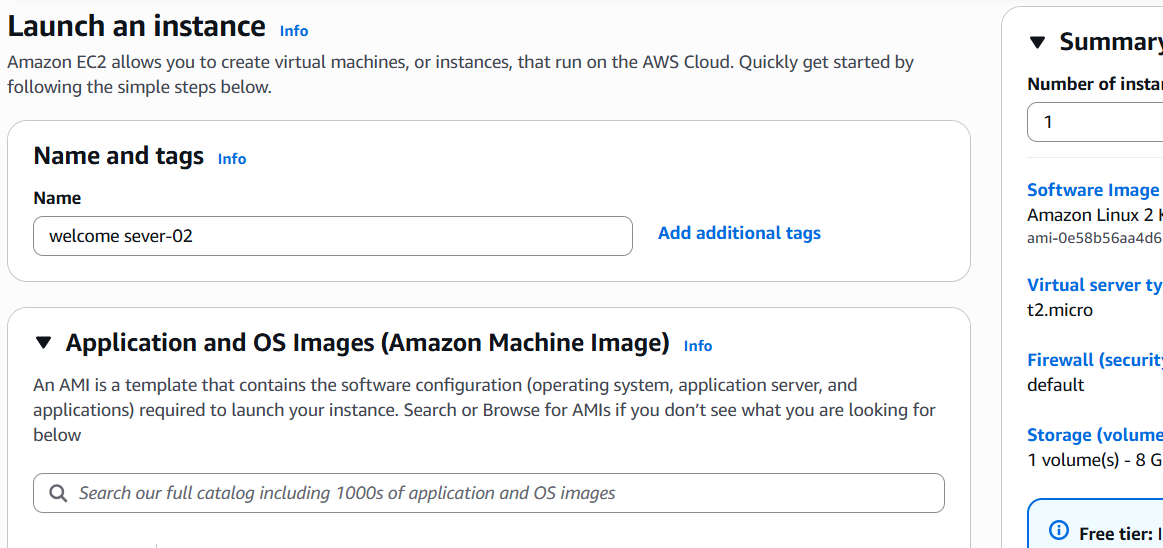
**Review and Create**:

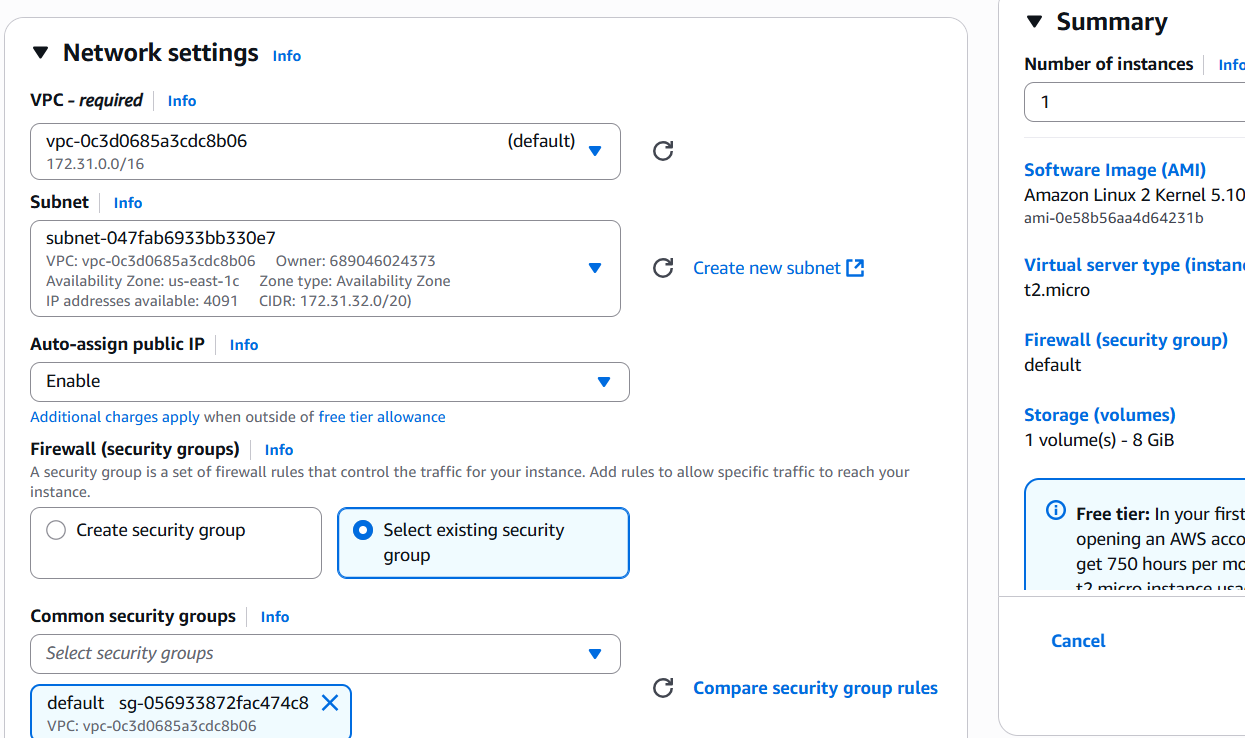
* Review all configurations.
* Click **Create** to launch the load balancer

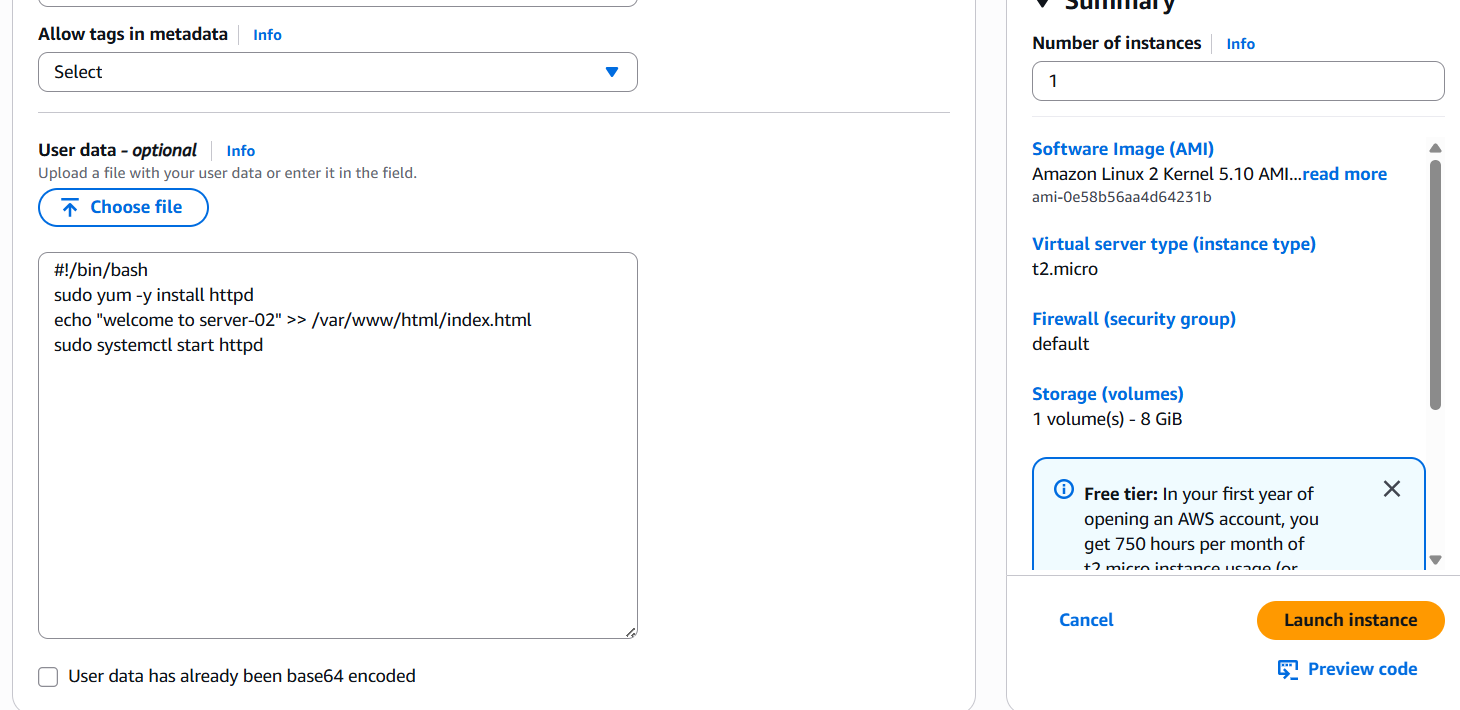




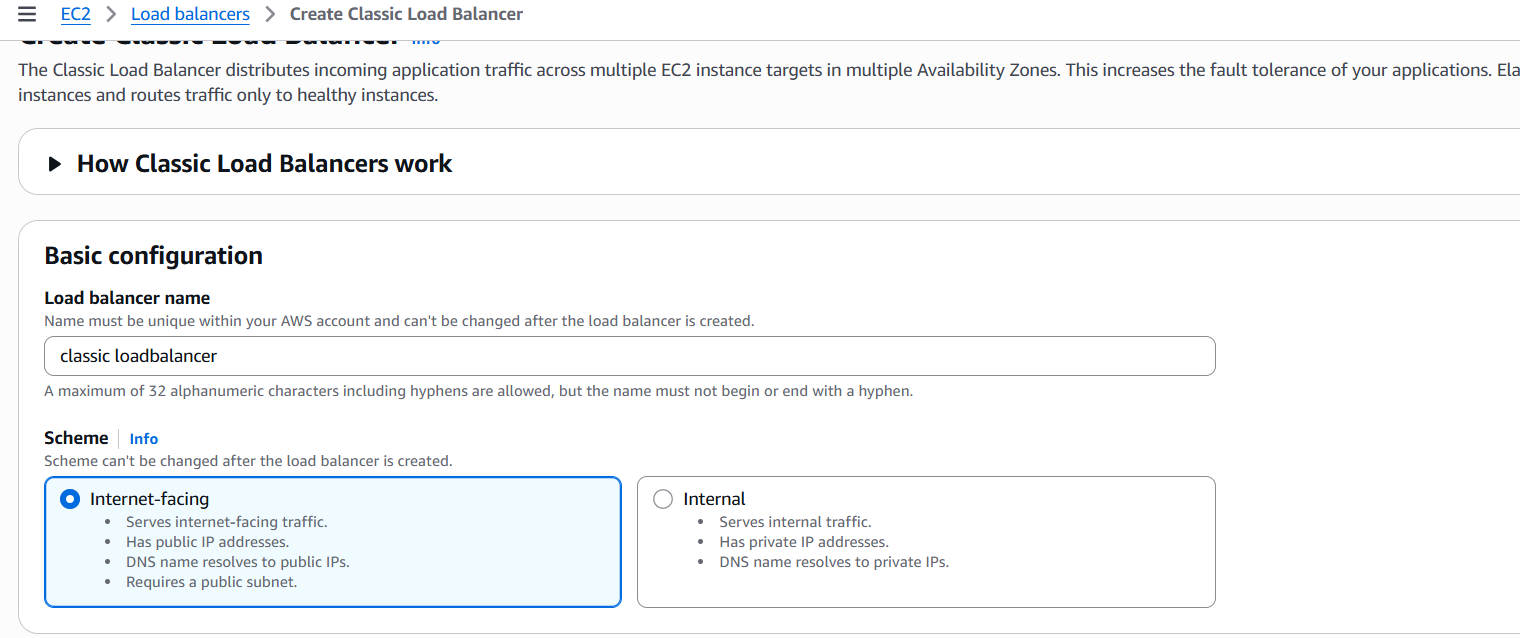


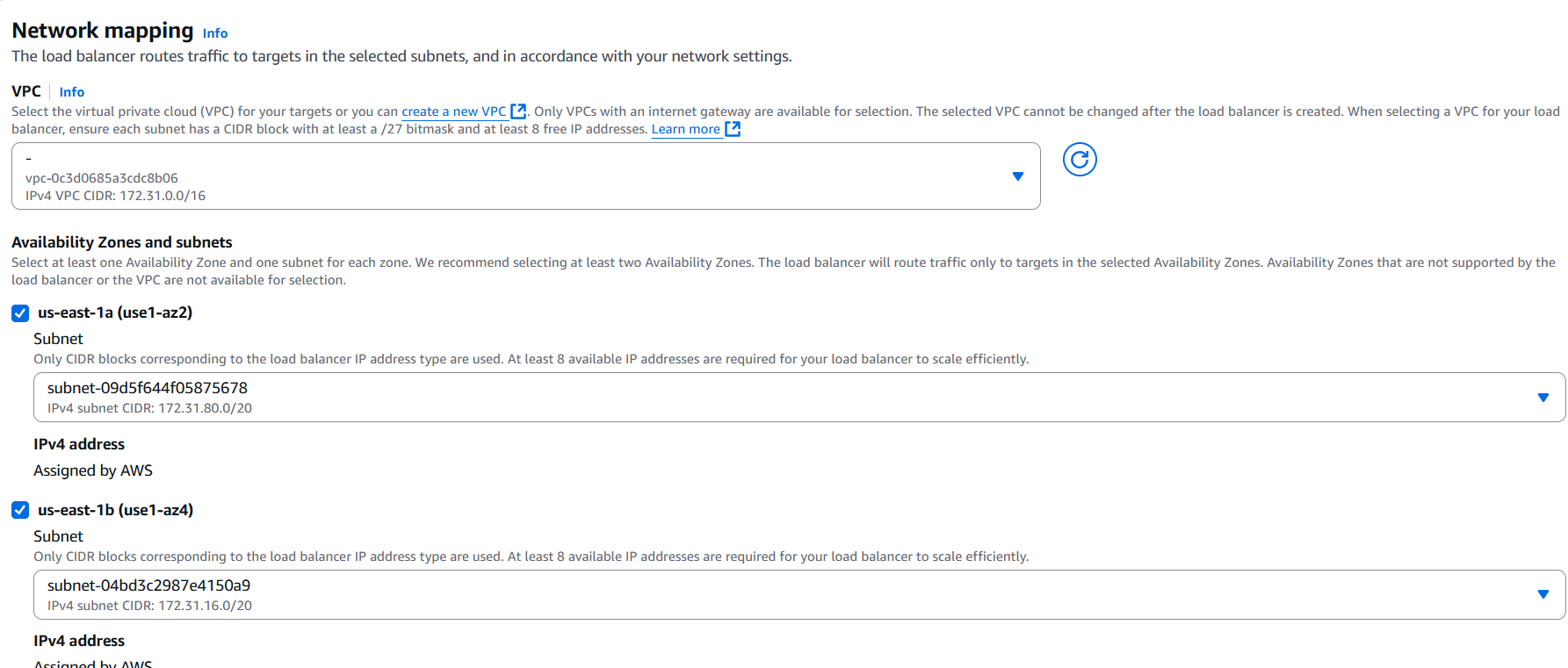


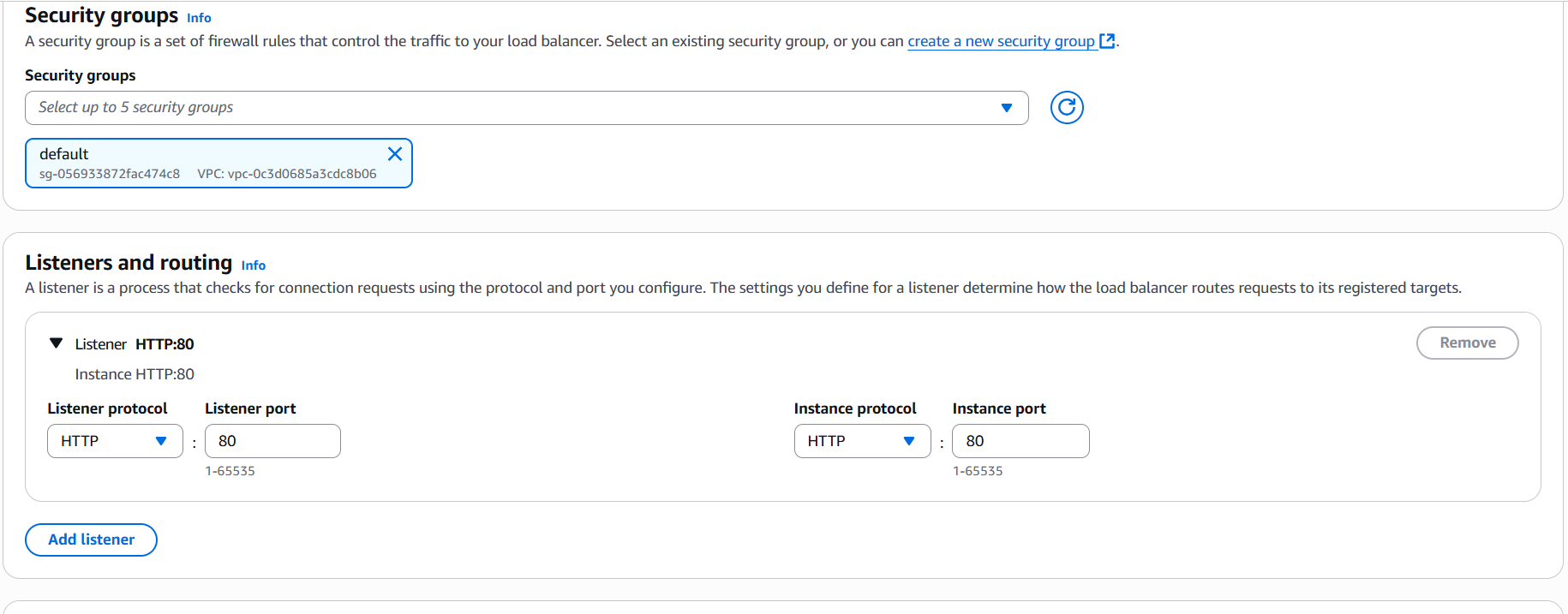


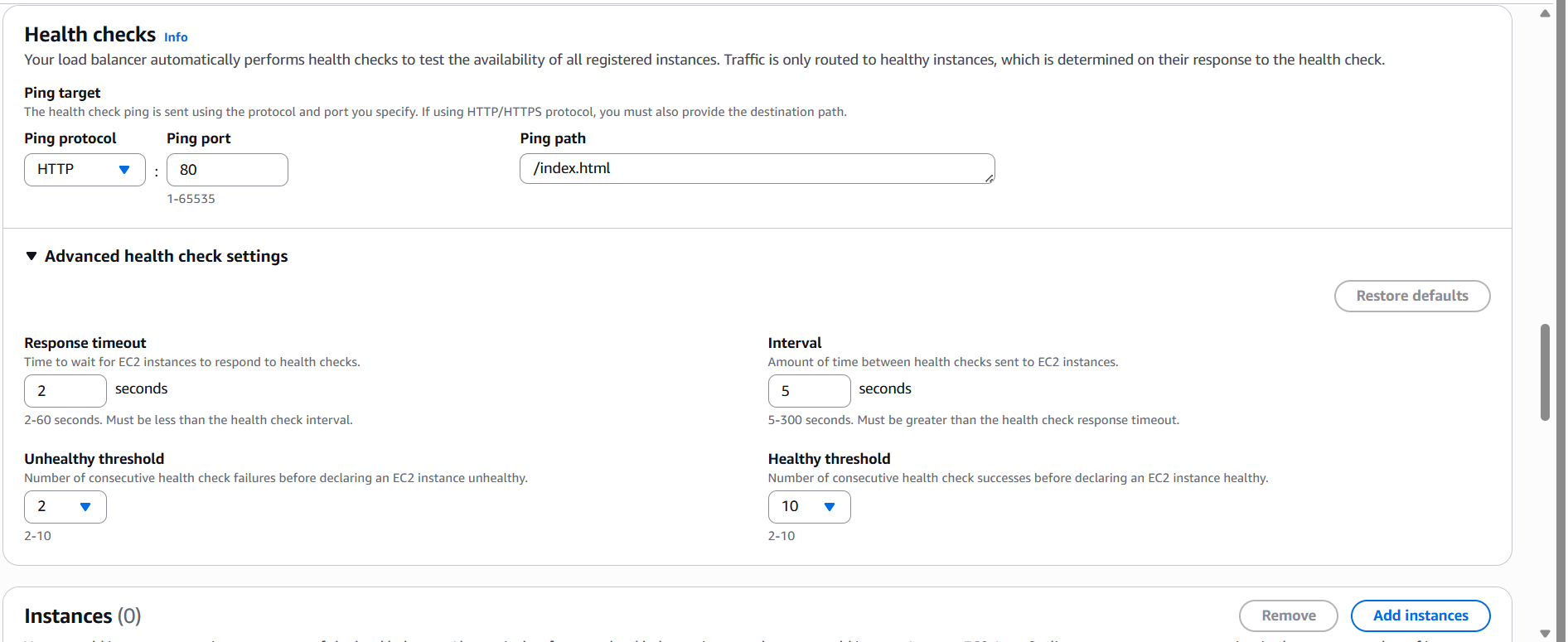


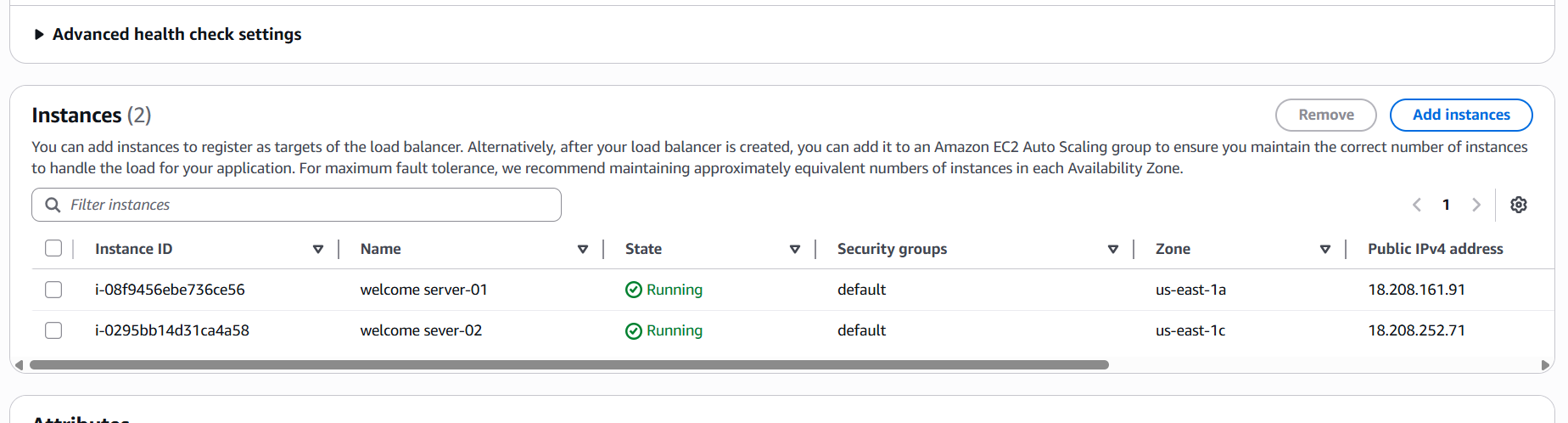


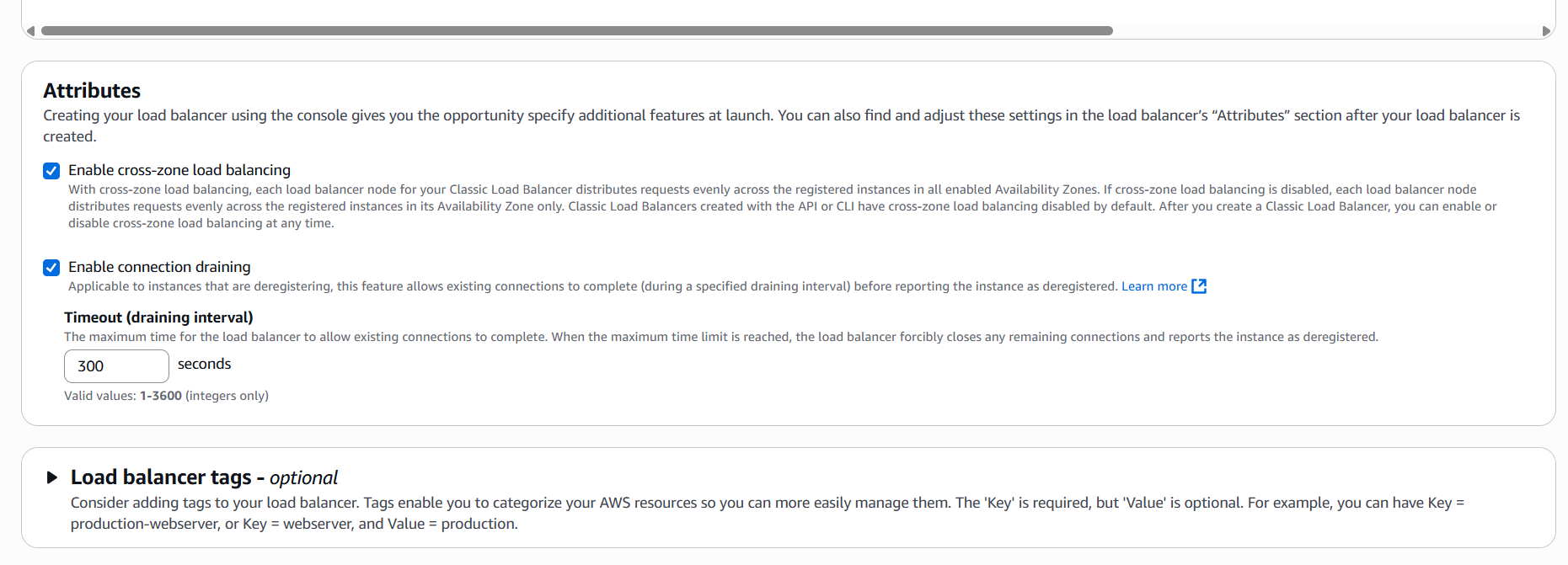


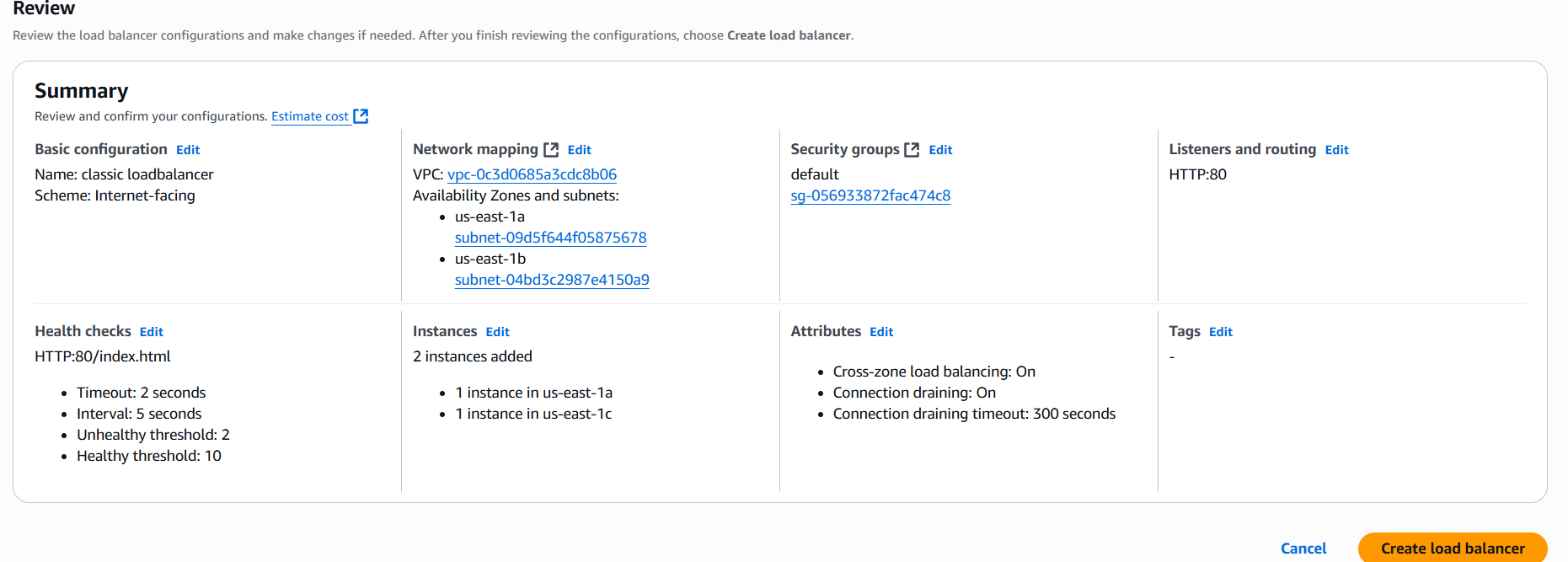


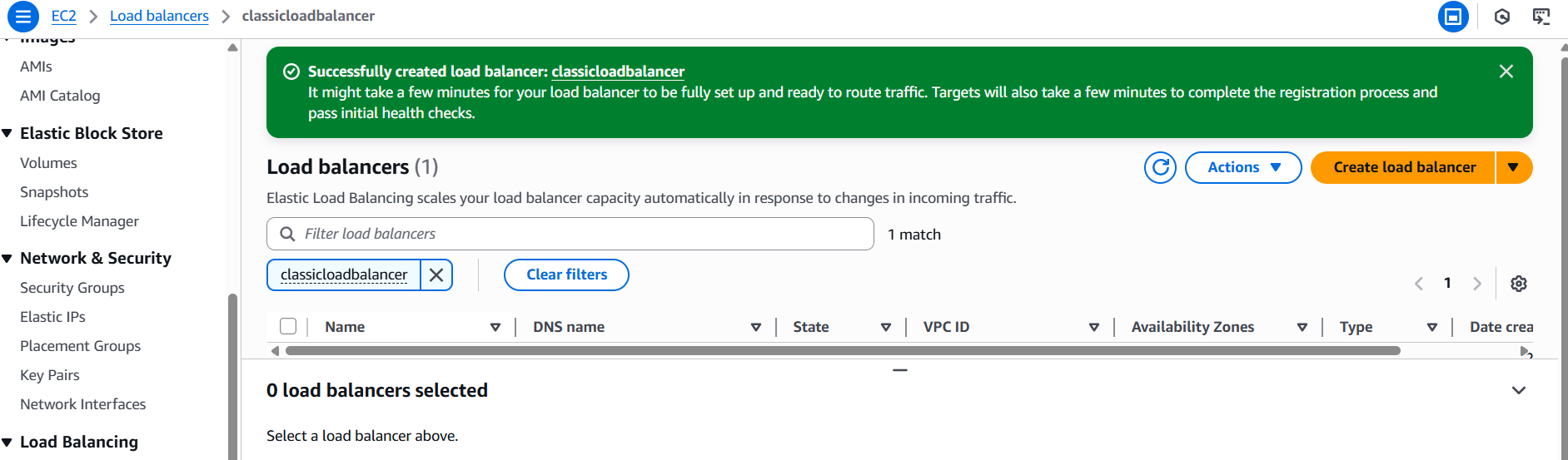












1. Configure Application Load balancer.

**Create a Target Group**

A target group routes requests to one or more registered targets (e.g., EC2 instances).

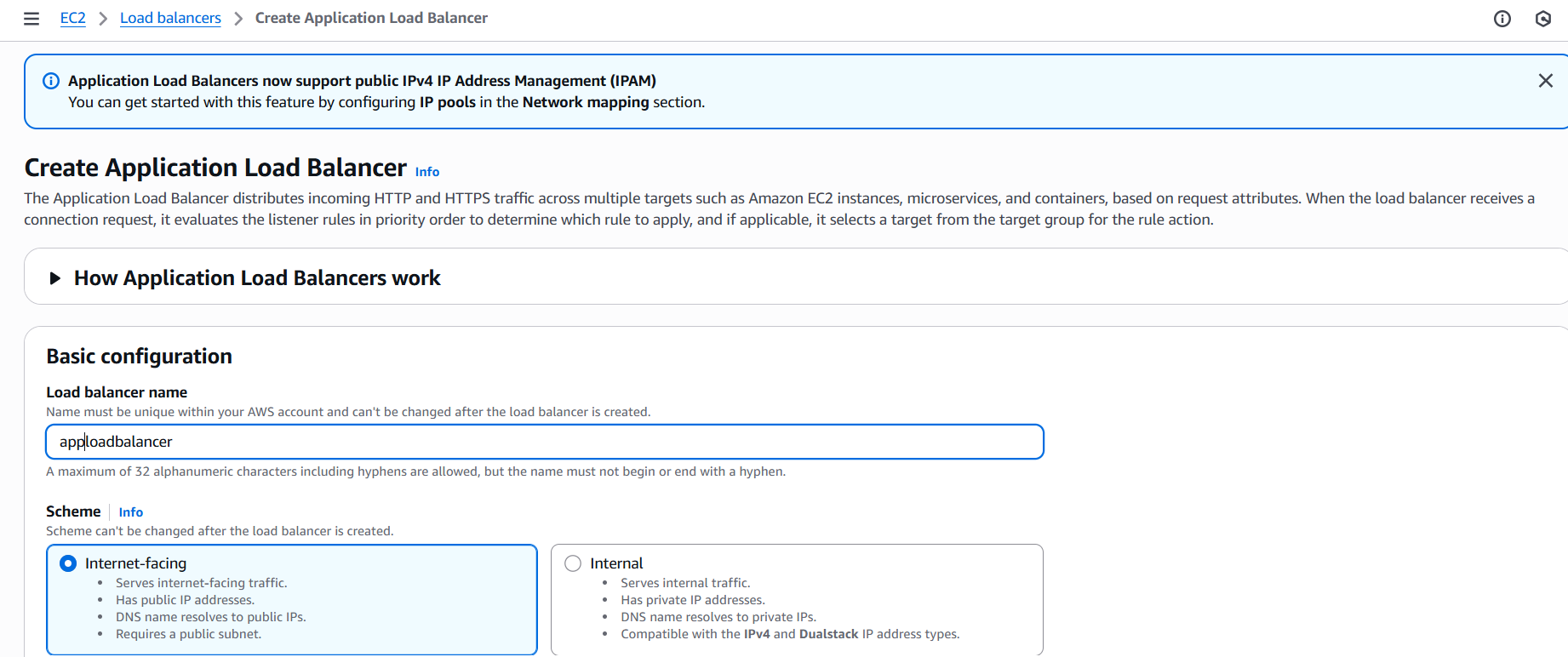
1. Navigate to the **EC2 Dashboard** in the AWS Management Console.
2. In the left-hand menu, under **Load Balancing**, select **Target Groups**.
3. Click **Create target group**.
4. Set the following configurations:
   * **Target type**: Instance
   * **Name**: Enter a name for the target group
   * **Protocol**: HTTP
   * **Port**: 80
   * **VPC**: Select the VPC containing your EC2 instances
5. Click **Next**.
6. On the **Register targets** page:
   * Select the EC2 instances you want to include.
   * Keep the default port (80).
   * Click **Include as pending below**.
7. Click **Create target group**.

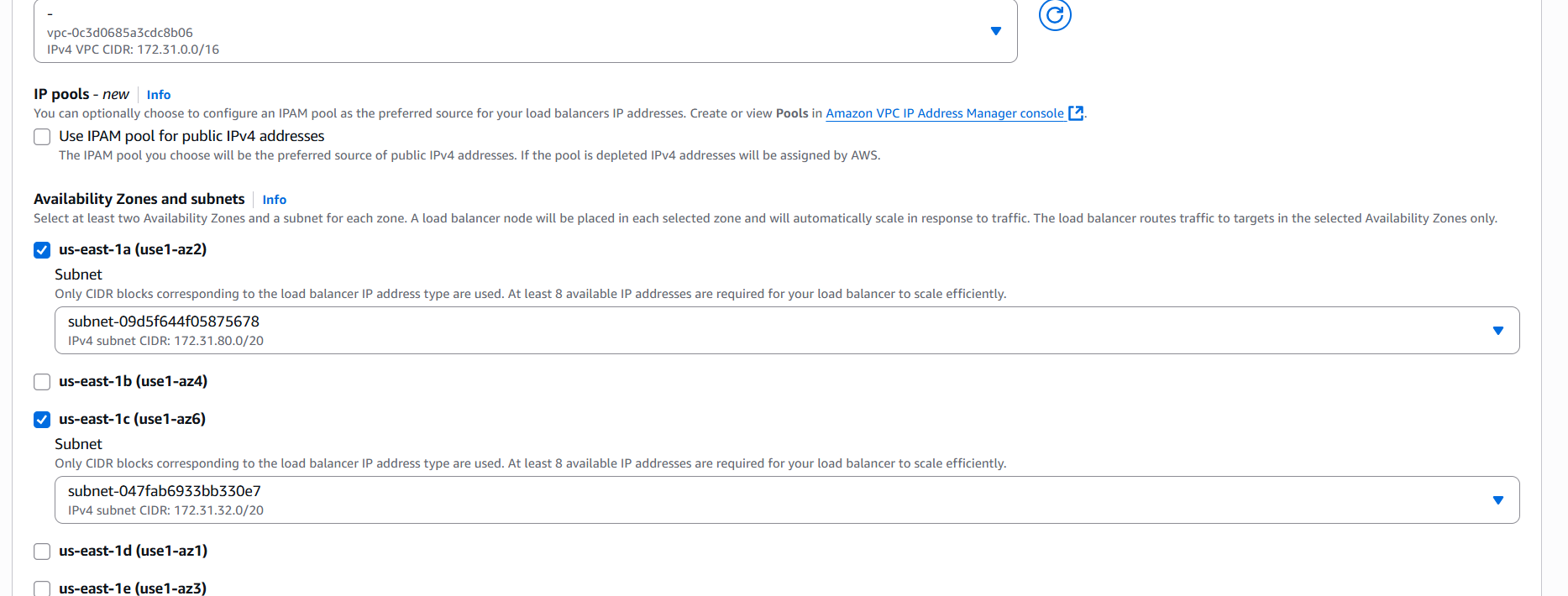
**Create the Application Load Balancer**

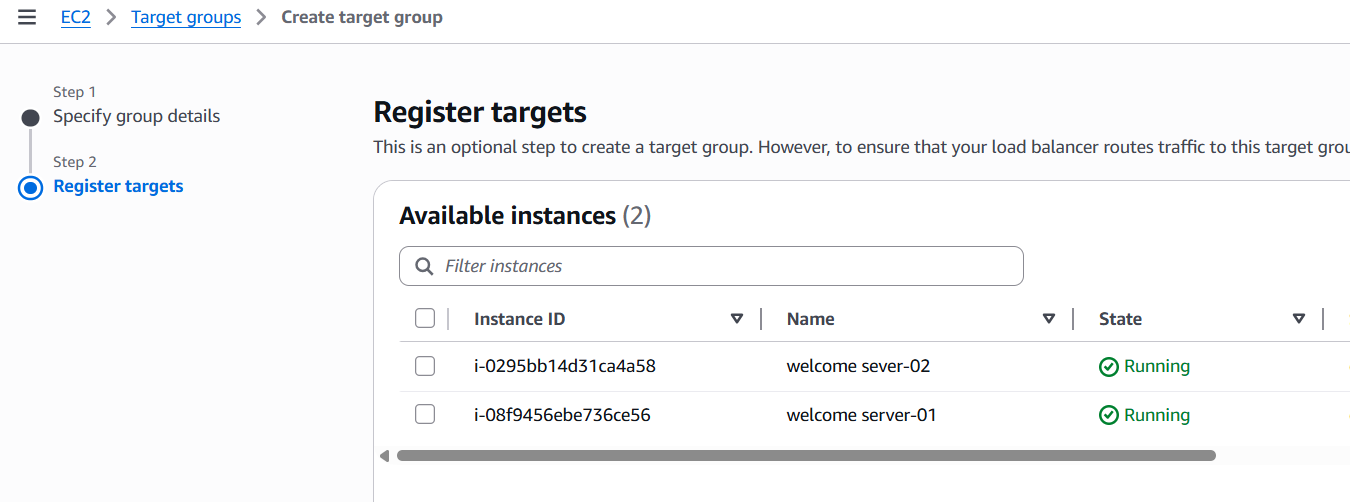
1. In the **EC2 Dashboard**, under **Load Balancing**, select **Load Balancers**.
2. Click **Create Load Balancer**.
3. Choose **Application Load Balancer** and click **Create**.
4. Configure the following settings:
   * **Name**: Enter a name for the load balancer
   * **Scheme**: Internet-facing (for public access) or Internal (for internal applications)
   * **IP address type**: IPv4 or Dualstack
5. Under **Network mapping**:
   * Select the VPC containing your EC2 instances.
   * Choose at least two Availability Zones and select one public subnet in each.
6. Under **Security groups**:
   * Select an existing security group or create a new one that allows inbound HTTP traffic on port 80.
7. Under **Listeners and routing**:
   * Listener protocol: HTTP
   * Port: 80
   * Default action: Select the target group created earlier
8. Click **Create load balancer**.

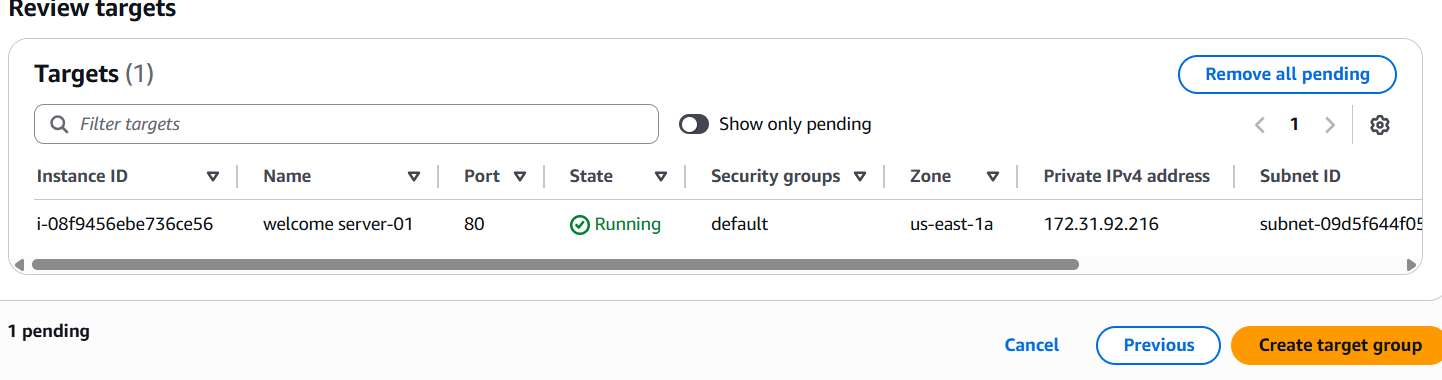
**Verify the Load Balancer**

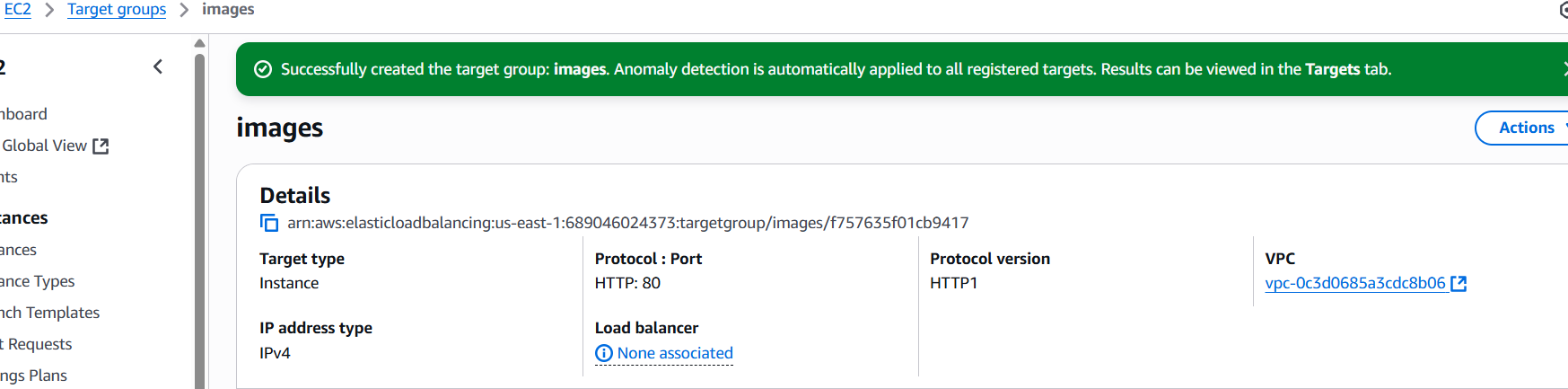
1. After the load balancer is created, navigate to **Load Balancers** in the EC2 Dashboard.
2. Select your newly created load balancer and note its **DNS name**.
3. Open a web browser and enter the DNS name.
4. You should see the default page of your web server, confirming that the load balancer is routing traffic to your EC2 instances.

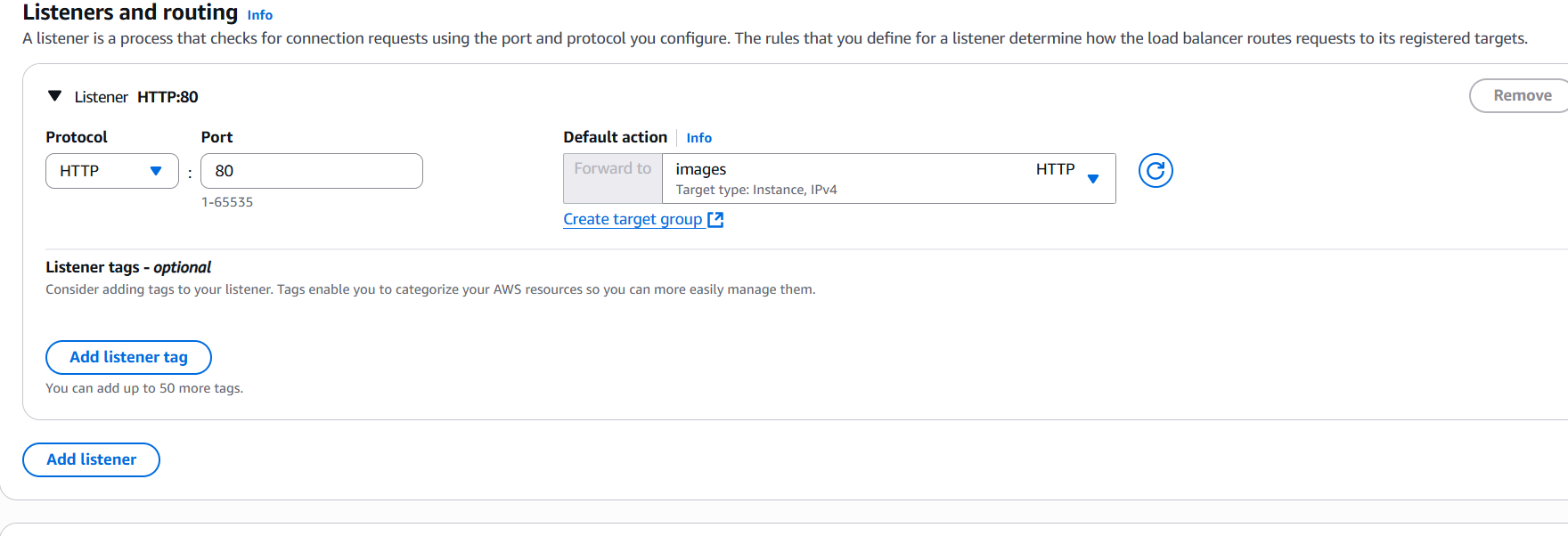


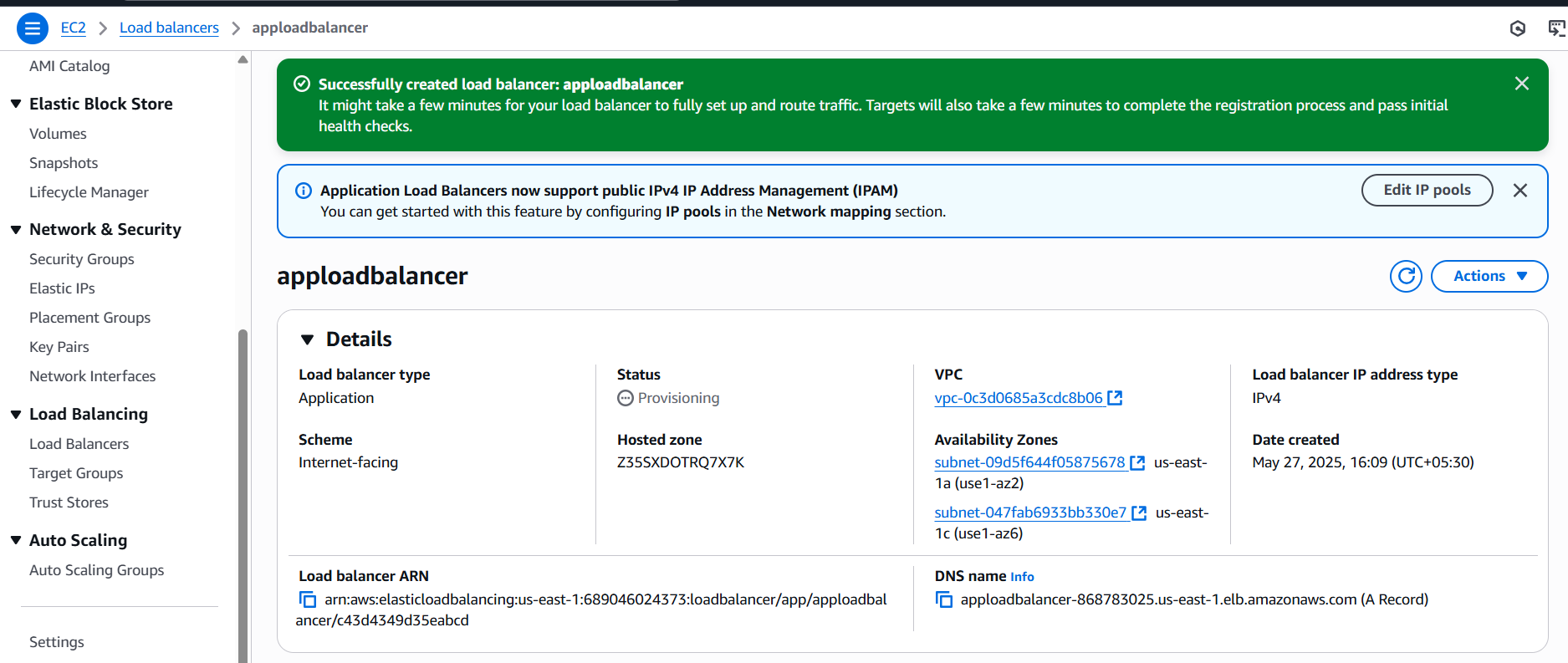


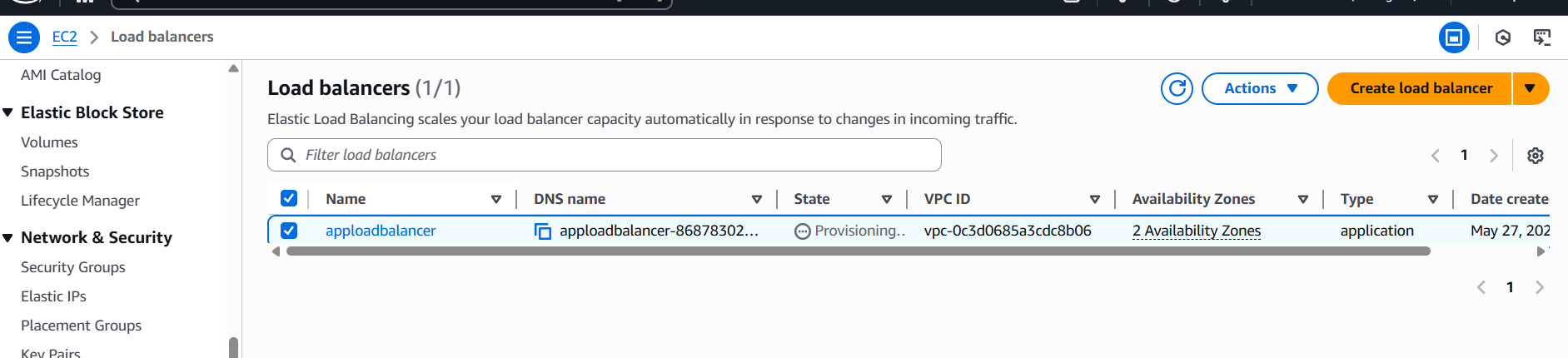


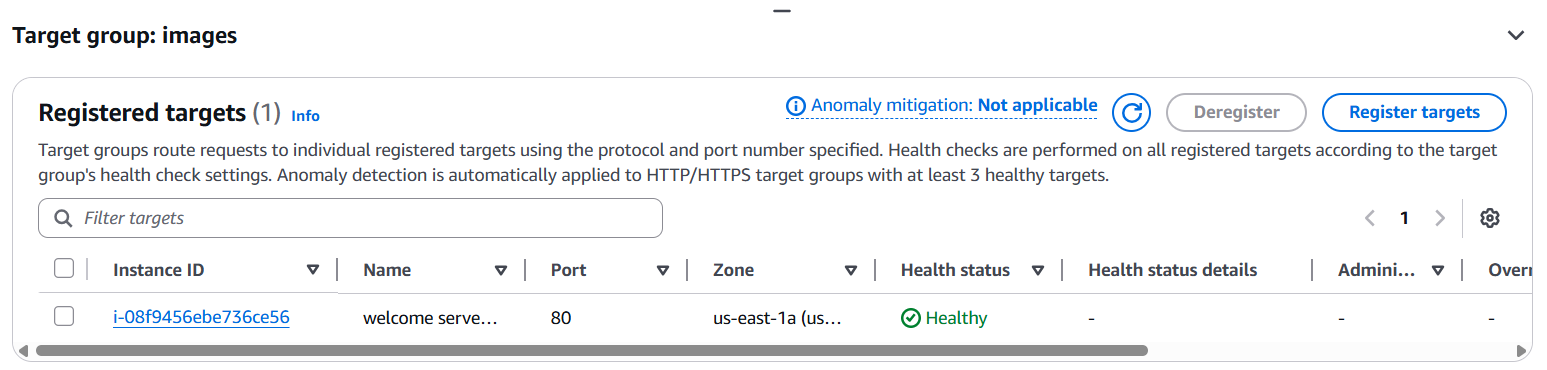


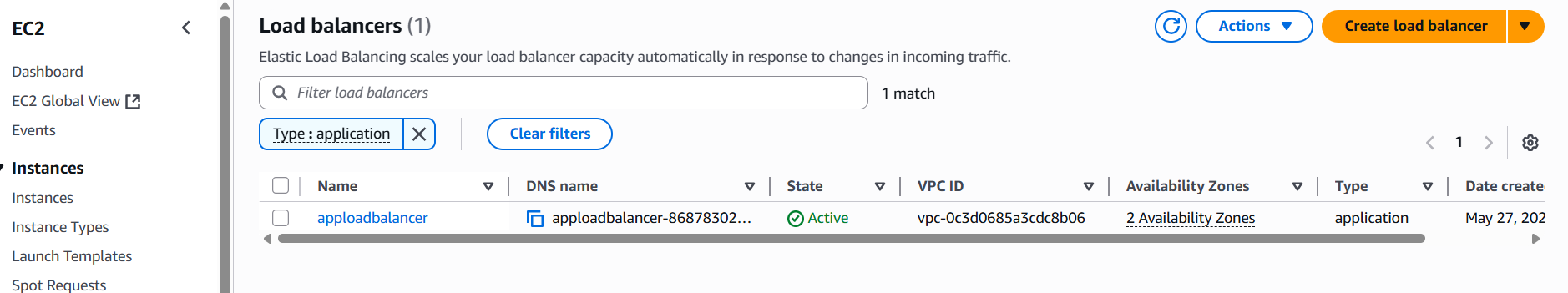


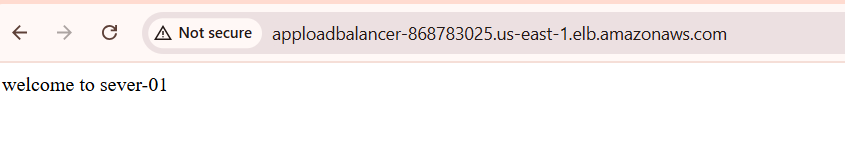


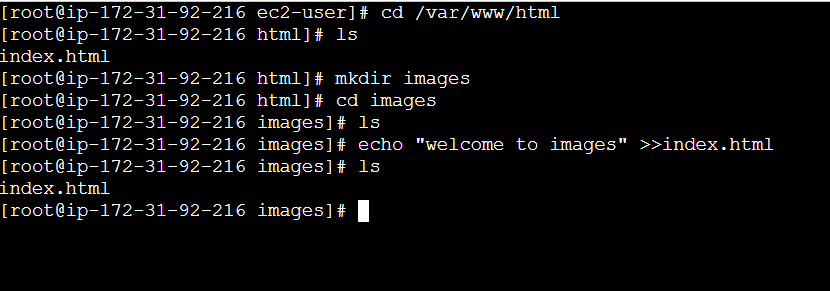


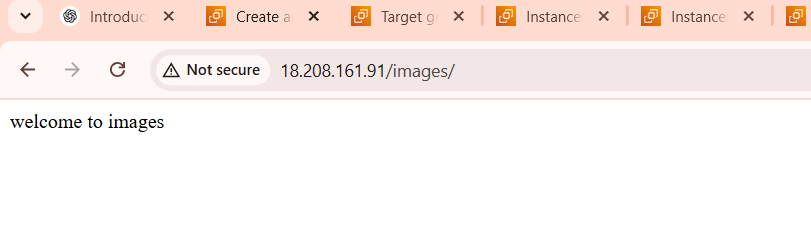












1. Configure Network Load balancer.

**Step 1: Prepare Your Environment**

Before setting up the NLB:

* **Launch EC2 Instances**: Ensure you have at least two EC2 instances running in different Availability Zones within the same Virtual Private Cloud (VPC).
* **Install Necessary Services**: Install and configure the services (e.g., web servers) on each instance to handle TCP or UDP traffic.
* **Security Groups**: Configure the security groups for your EC2 instances to allow inbound traffic on the necessary ports (e.g., TCP port 80).

**Step 2: Create a Target Group**

A target group routes requests to one or more registered targets (e.g., EC2 instances).

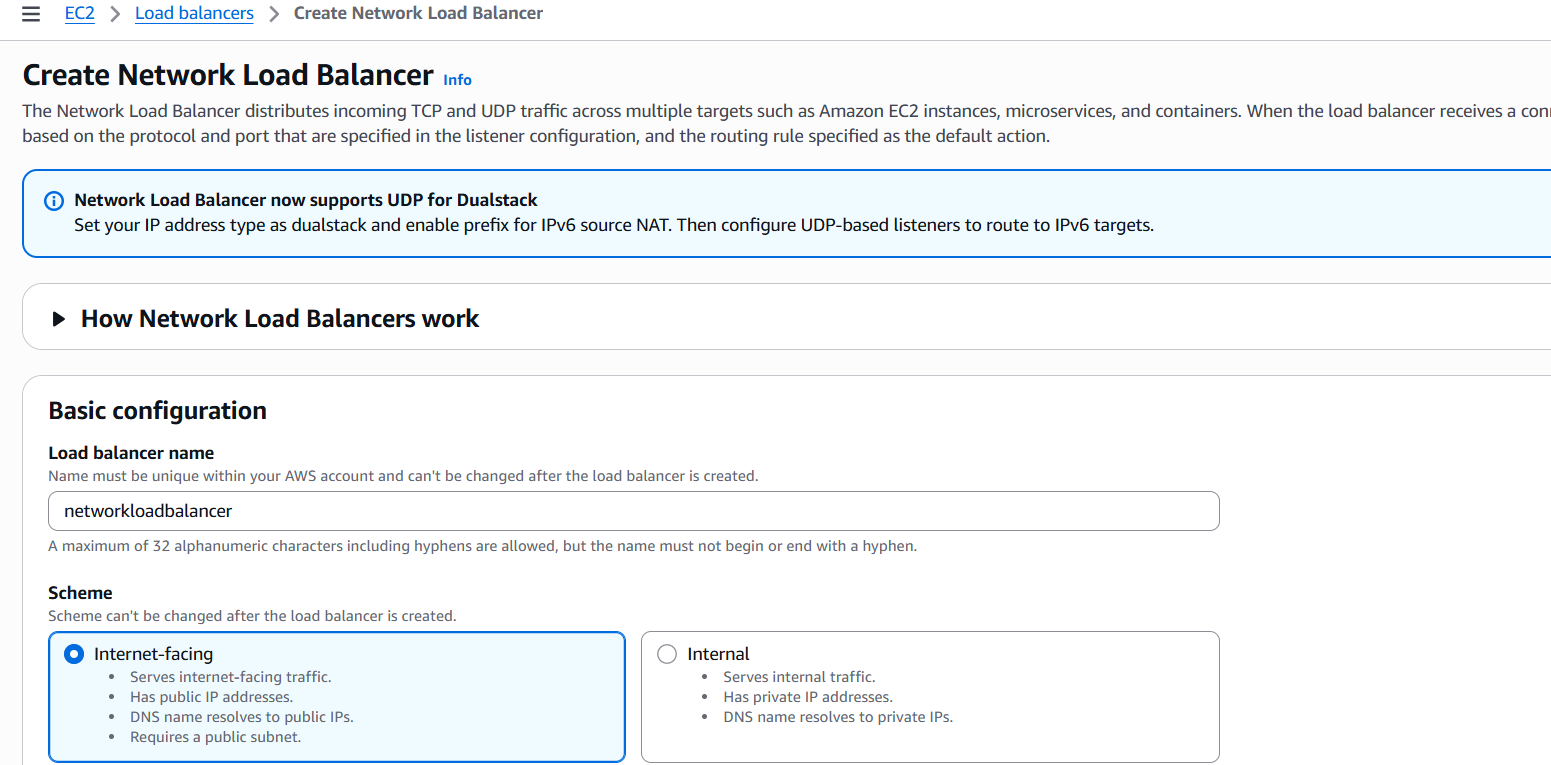
1. Navigate to the **EC2 Dashboard** in the AWS Management Console.
2. In the left-hand menu, under **Load Balancing**, select **Target Groups**.
3. Click **Create target group**.
4. Set the following configurations:
   * **Target type**: Instance
   * **Name**: Enter a name for the target group
   * **Protocol**: TCP
   * **Port**: 80
   * **VPC**: Select the VPC containing your EC2 instances
5. Click **Next**.
6. On the **Register targets** page:
   * Select the EC2 instances you want to include.
   * Keep the default port (80).
   * Click **Include as pending below**.
7. Click **Create target group**.

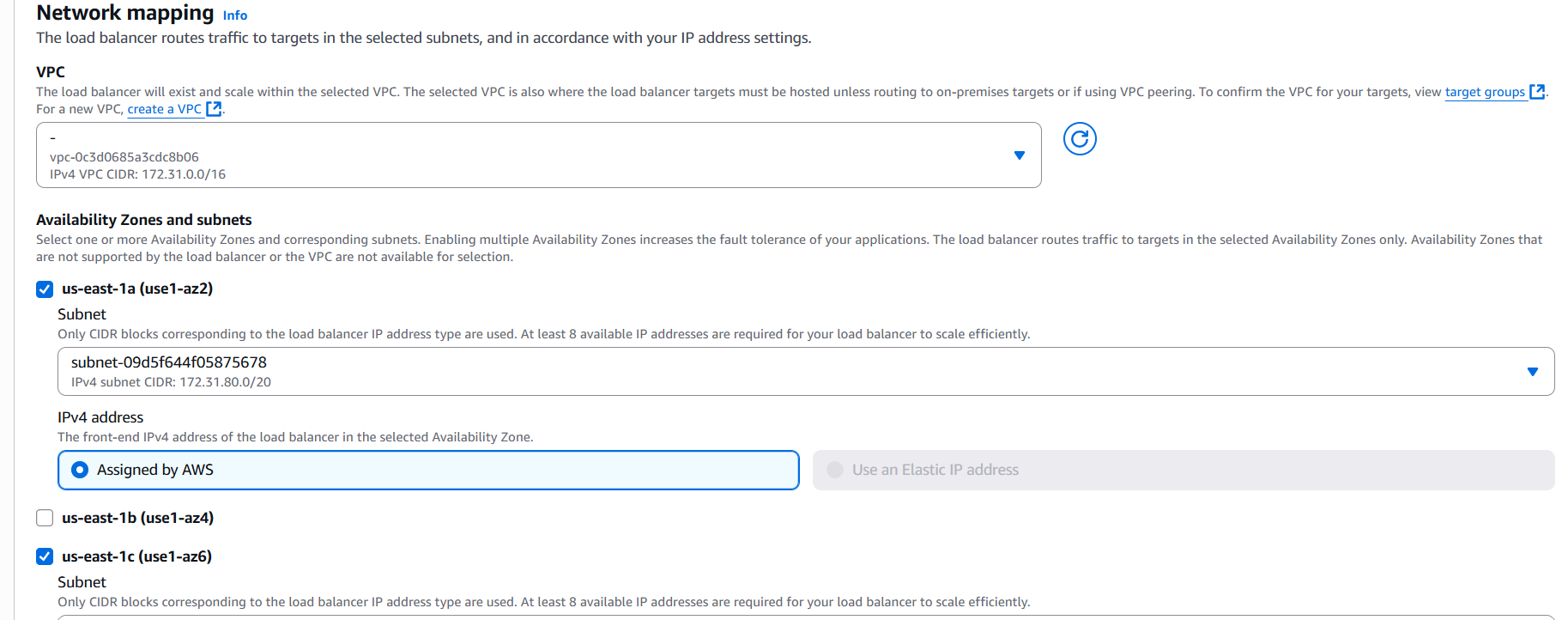
**Step 3: Create the Network Load Balancer**

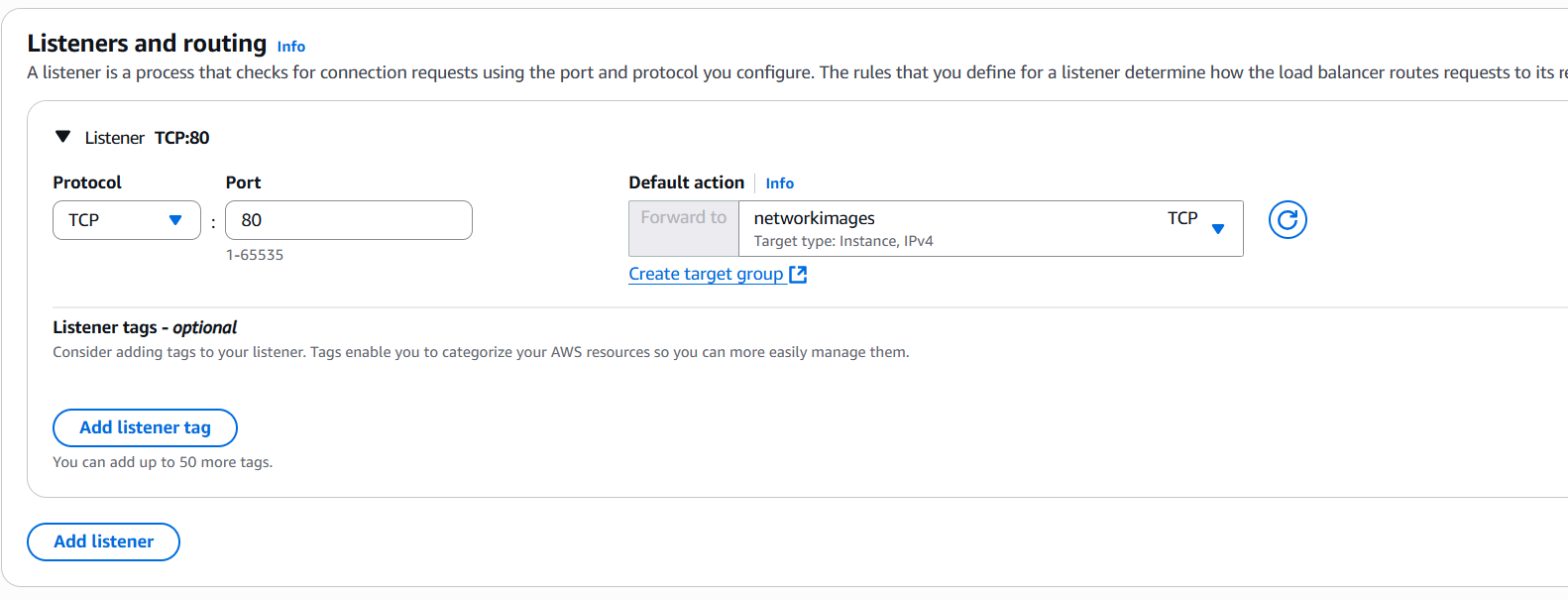
1. In the **EC2 Dashboard**, under **Load Balancing**, select **Load Balancers**.
2. Click **Create Load Balancer**.
3. Choose **Network Load Balancer** and click **Create**.
4. Configure the following settings:
   * **Name**: Enter a name for the load balancer
   * **Scheme**: Internet-facing (for public access) or Internal (for internal applications)
   * **IP address type**: IPv4 or Dualstack
5. Under **Network mapping**:
   * Select the VPC containing your EC2 instances.
   * Choose at least two Availability Zones and select one public subnet in each.
6. Under **Listeners and routing**:
   * Listener protocol: TCP
   * Port: 80
   * Default action: Select the target group created earlier
7. Click **Create load balancer**.

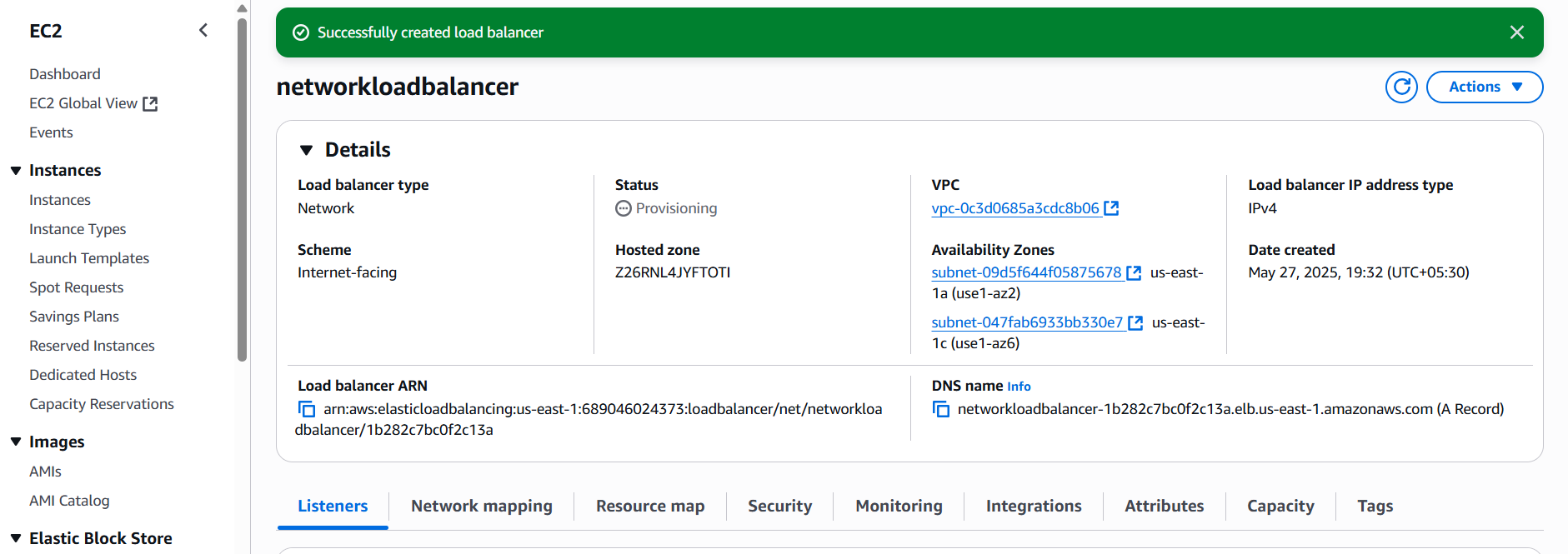
**Step 4: Verify the Load Balancer**

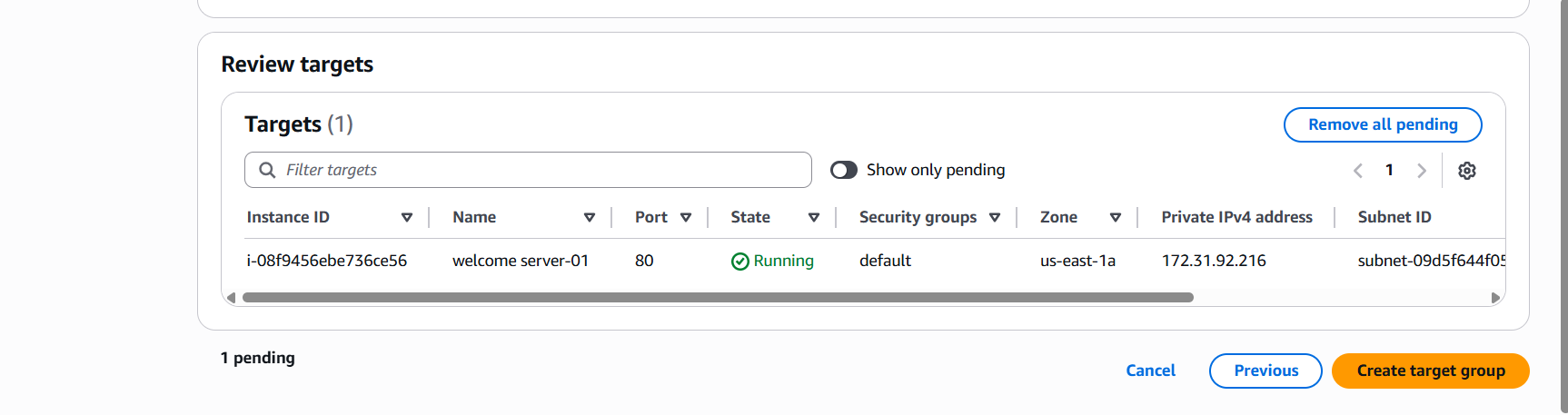
1. After the load balancer is created, navigate to **Load Balancers** in the EC2 Dashboard.
2. Select your newly created load balancer and note its **DNS name**.
3. Open a web browser and enter the DNS name.
4. You should see the default page of your web server, confirming that the load balancer is routing traffic to your EC2 instances.

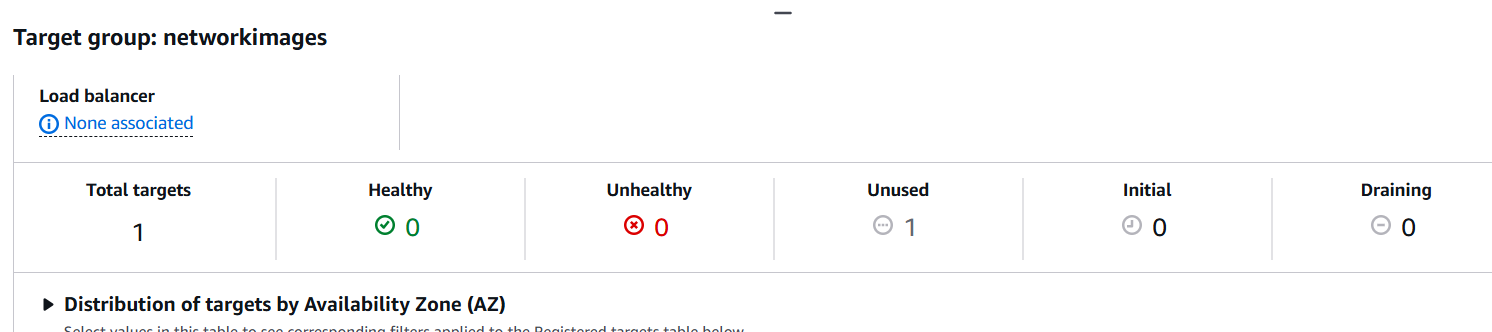


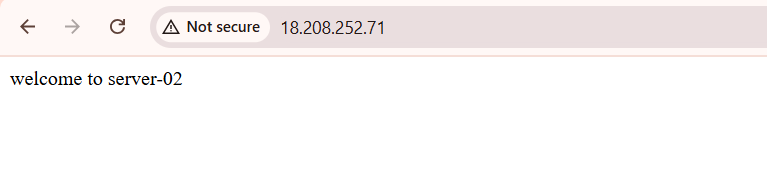












1. Attach SSL for application load balancer.
2. **Access ACM Console:**
   * Navigate to the [AWS Certificate Manager (ACM) console](https://console.aws.amazon.com/acm/home).
3. **Request a Public Certificate:**
   * Click on **"Request a certificate"**.
   * Choose **"Request a public certificate"** and click **"Next"**.
   * Enter your domain name (e.g., example.com or www.example.com).
   * Click **"Next"**.
4. **Validate Domain Ownership:**
   * Select a validation method: **DNS validation** or **Email validation**.
   * Follow the on-screen instructions to complete the validation process.
5. **Review and Request:**
   * Review your request and click **"Confirm and request"**.
   * After validation, the certificate status will change to **"Issued"**.

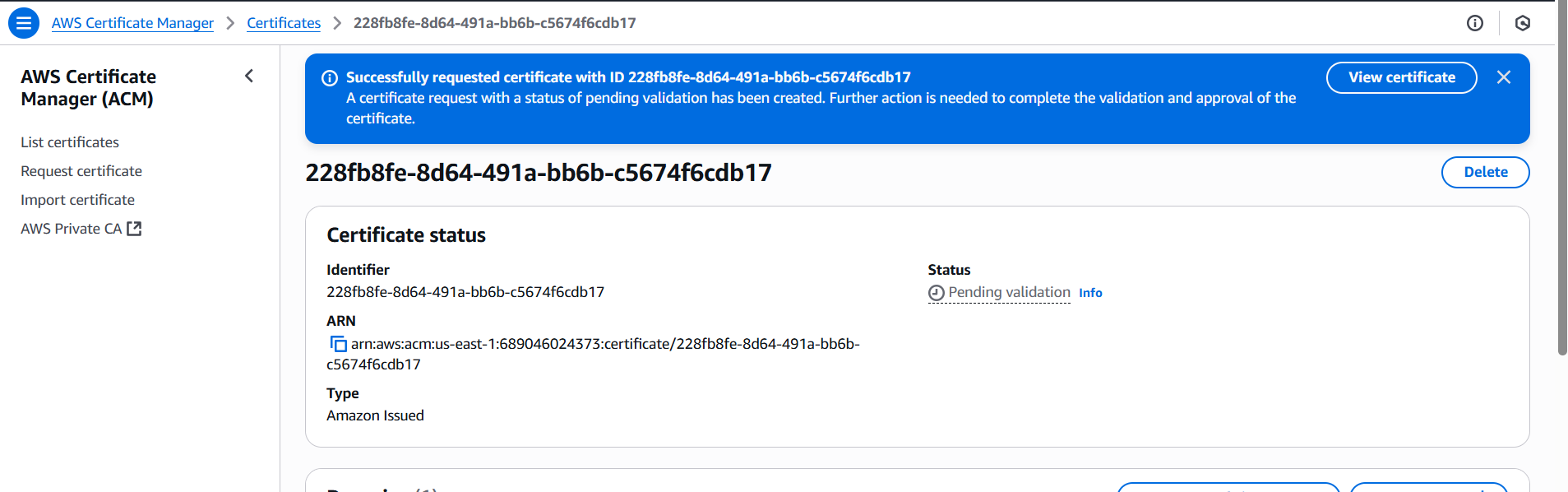
**Step 2: Add an HTTPS Listener to Your ALB**

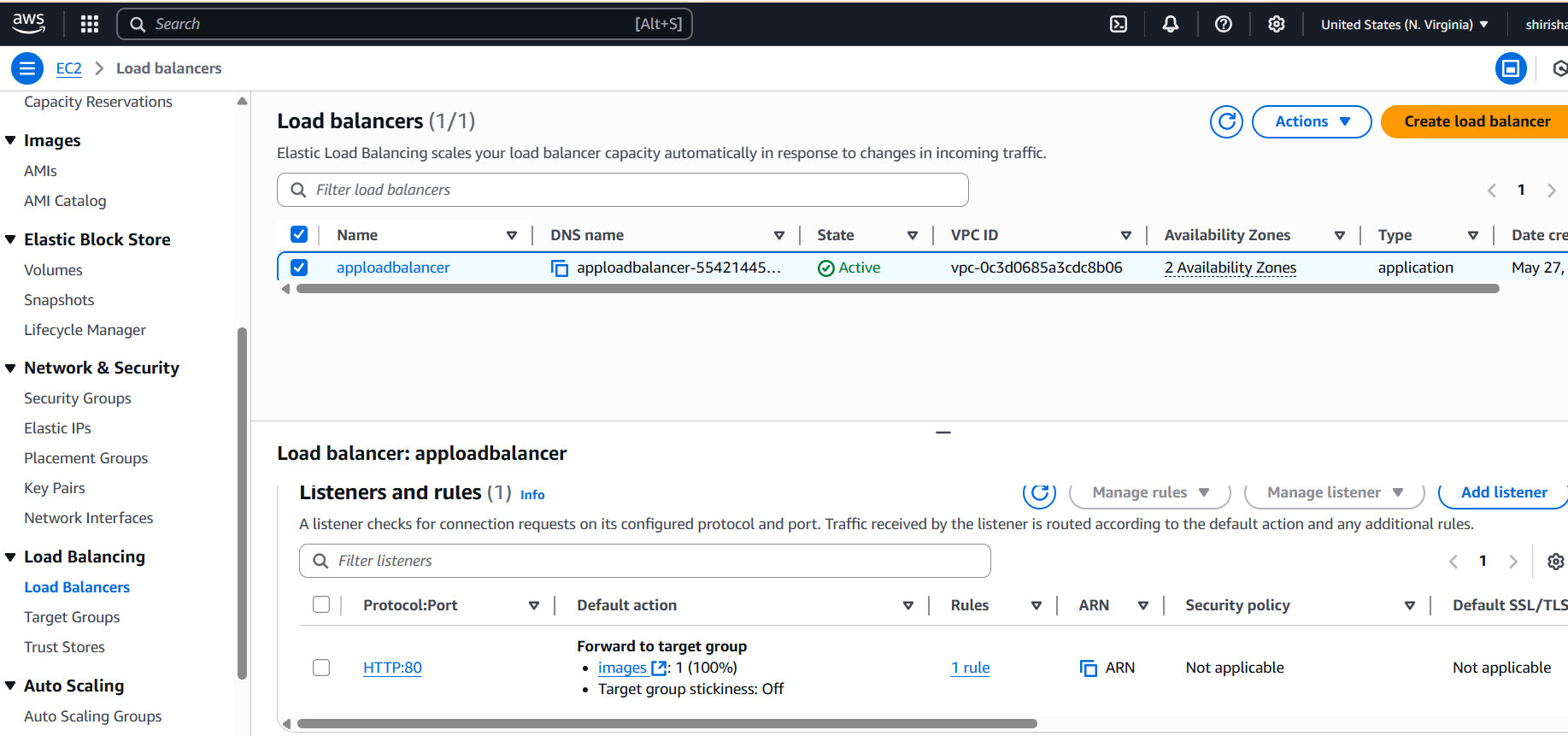
1. **Access EC2 Console:**
   * Navigate to the [Amazon EC2 console](https://console.aws.amazon.com/ec2/).[AWS Documentation+3AWS Documentation+3AWS Documentation+3](https://docs.aws.amazon.com/elasticloadbalancing/latest/application/create-https-listener.html?utm_source=chatgpt.com)
2. **Navigate to Load Balancers:**
   * In the left navigation pane, under **"Load Balancing"**, click on **"Load Balancers"**.
   * Select your existing ALB from the list.
3. **Add HTTPS Listener:**
   * Go to the **"Listeners"** tab.
   * Click on **"Add listener"**.
   * For **Protocol**, select **HTTPS**.
   * For **Port**, enter **443**.
   * Under **Default actions**, choose **"Forward to"** and select your target group.
   * Click **"Add"** to proceed.
4. **Configure SSL Certificate:**
   * In the **"Secure listener settings"** section, under **"Default SSL certificate"**, choose **"From ACM"**.
   * Select the certificate you obtained earlier.
   * Choose a security policy (e.g., **ELBSecurityPolicy-2016-08**).
   * Click **"Add"** to finalize the listener configuration.

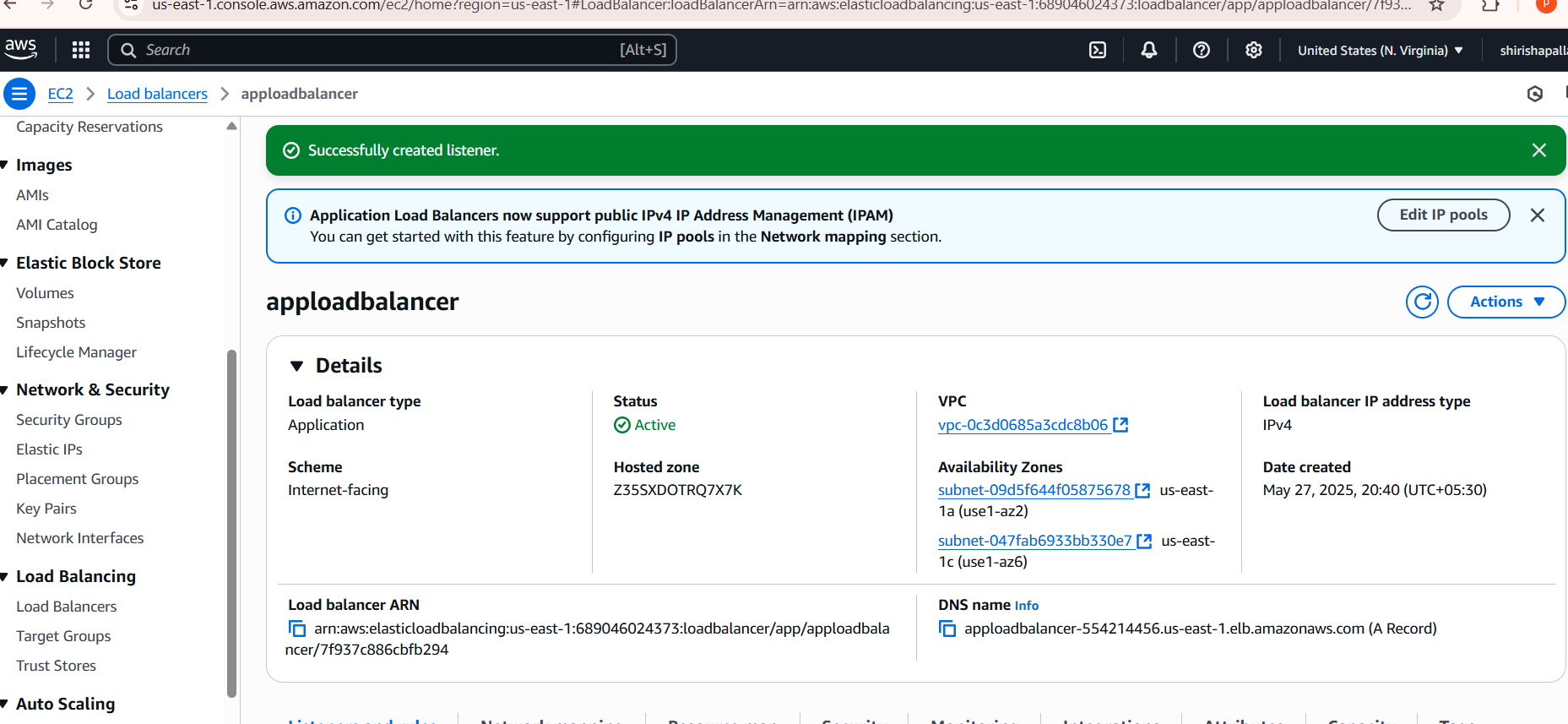
**Step 3: (Optional) Redirect HTTP to HTTPS**

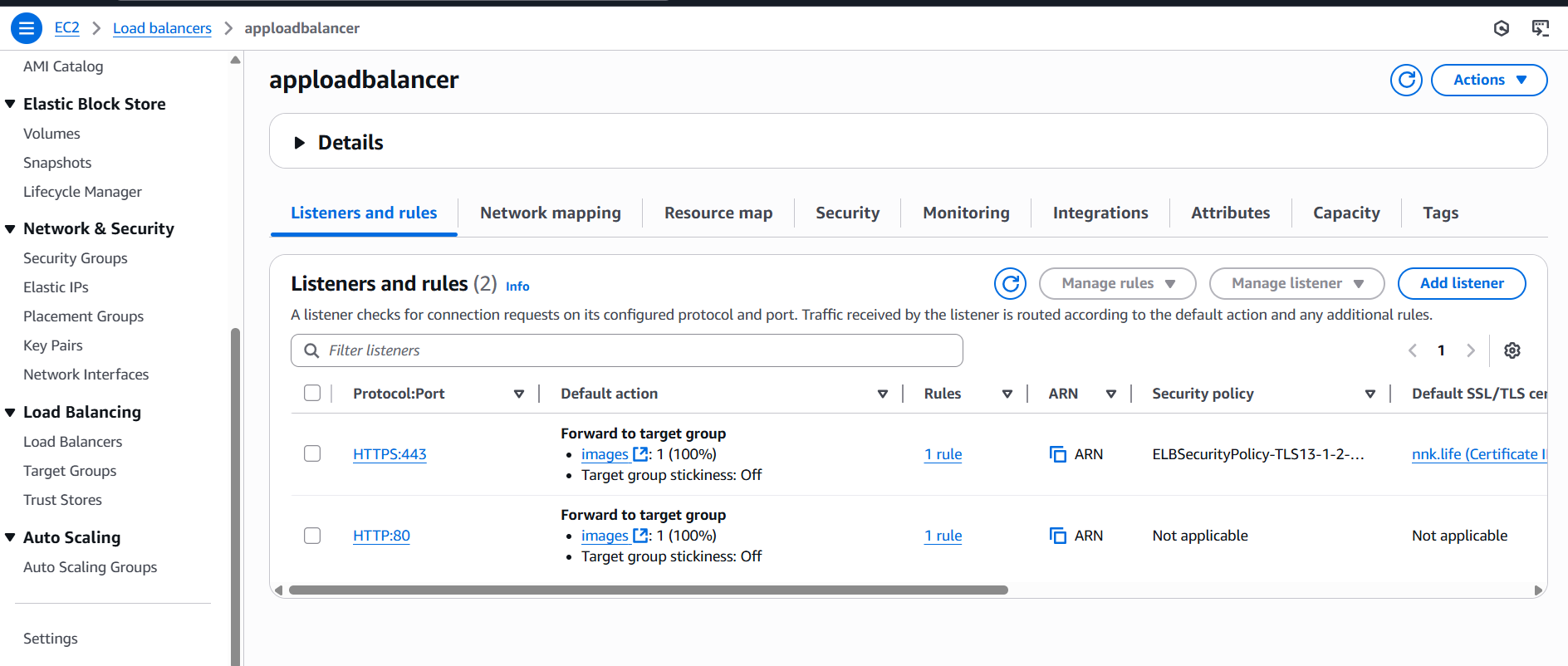
To ensure all traffic uses HTTPS, you can set up a redirect from HTTP to HTTPS:

1. **Add HTTP Listener:**
   * In the **"Listeners"** tab, click on **"Add listener"**.
   * For **Protocol**, select **HTTP**.
   * For **Port**, enter **80**.
   * Under **Default actions**, choose **"Redirect to"** and configure the redirect settings:
     + **Protocol:** HTTPS
     + **Port:** 443
     + **Status code:** 301
   * Click **"Add"** to implement the redirect









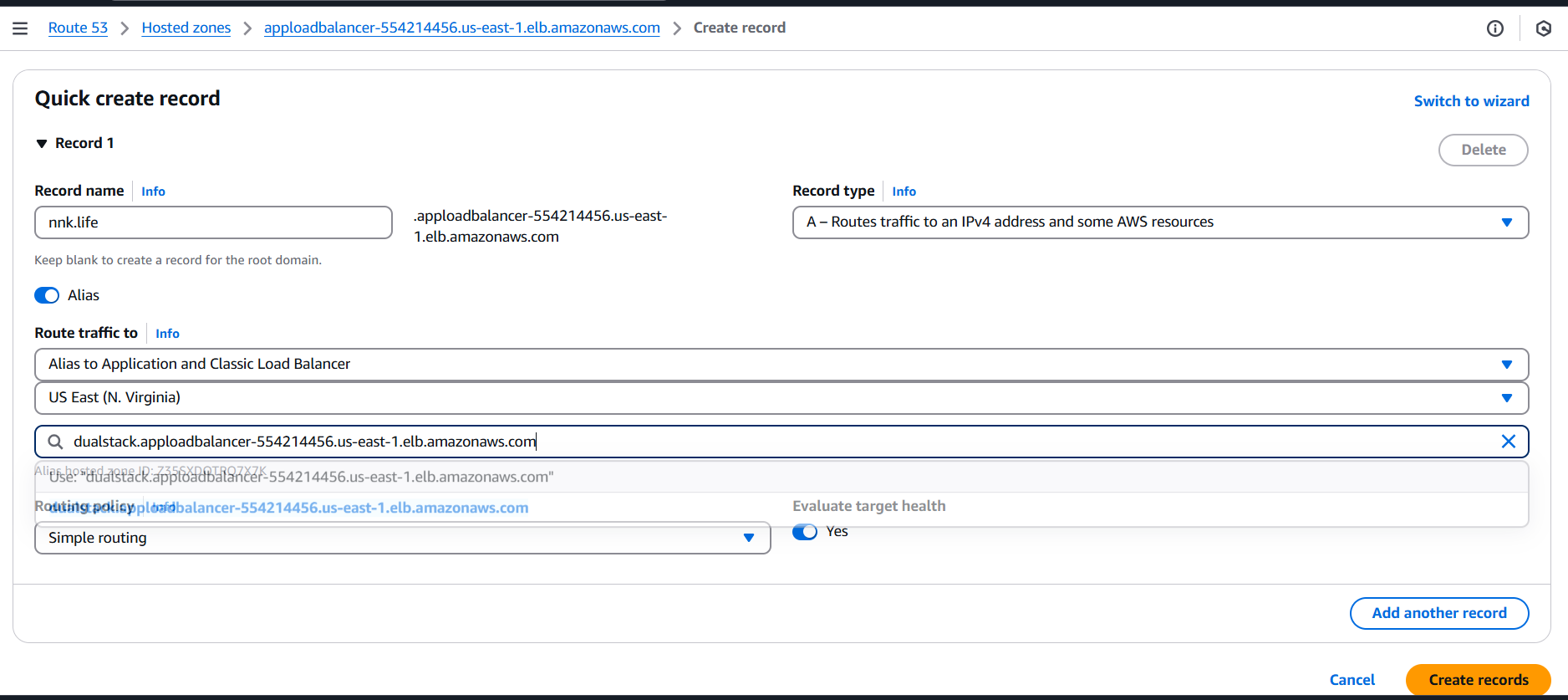
5) Map Applciation load balancer to R53.

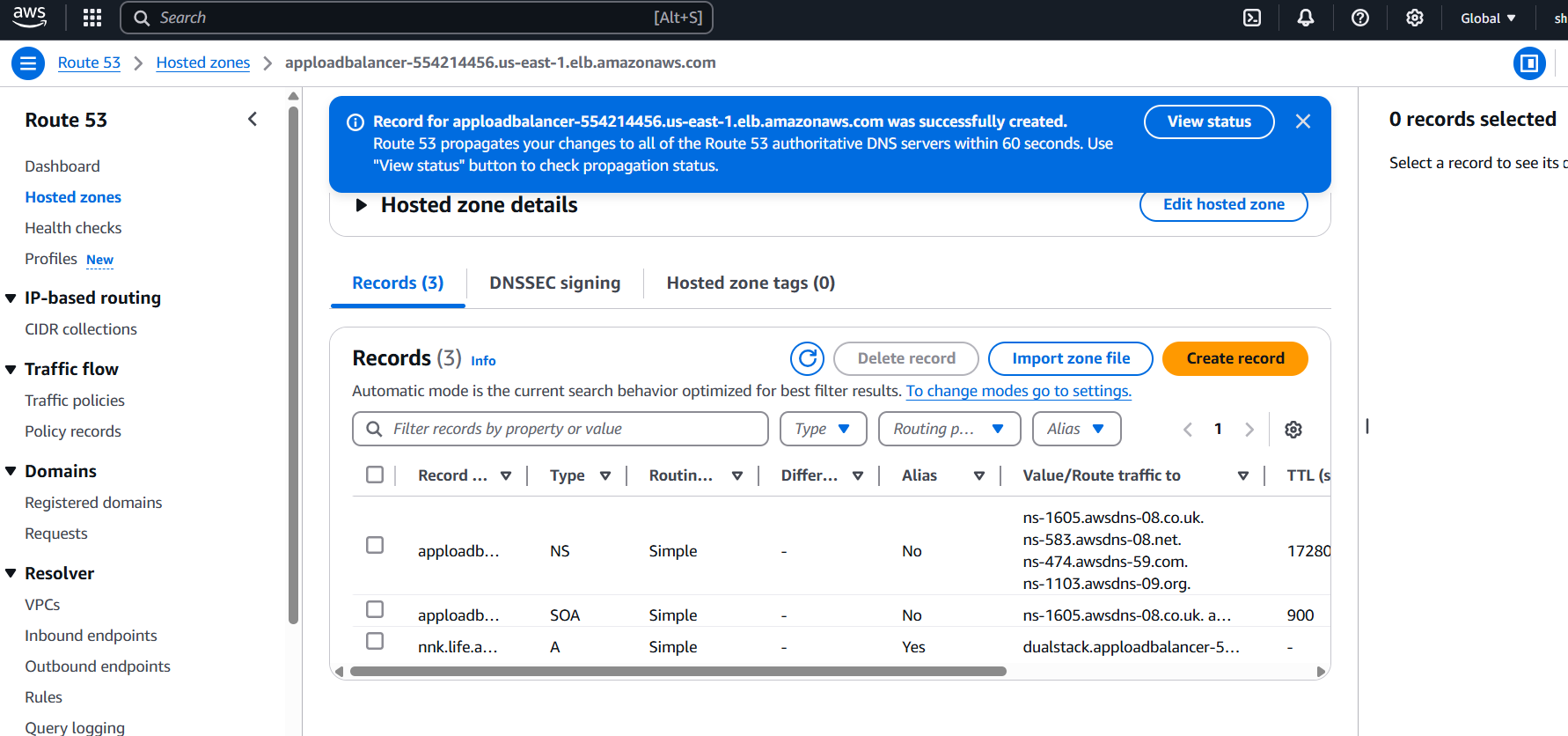
**Access Route 53 Console:**

* Navigate to the **Route 53 Console:**
* **Navigate to Hosted Zones:**
* In the left navigation pane, click on **"Hosted zones"**.
* Select the hosted zone corresponding to your domain name.

**Create a Record Set:**

* Click on **"Create record"**.
* For **Record name**, enter the subdomain you wish to use (e.g., www). Leave it blank if you're setting up the root domain.
* Set **Record type** to **A – IPv4 address**.
* Toggle **Alias** to **Yes**.
* For **Alias to**, choose **Application and Classic Load Balancer**.
* Select the appropriate **Region** where your ALB is deployed.
* In the **Load balancer** field, select your ALB from the list.
* Optionally, set **Evaluate target health** to **Yes** if you want Route 53 to route traffic only to healthy targets.
* Click **"Create records"** to save





6) Push the application load balancer logs to s3.

**1. Create an S3 Bucket**

* **Region**: Ensure the bucket is in the same AWS Region as your ALB.
* **Encryption**: Use Amazon S3-managed keys (SSE-S3) for server-side encryption.
* **Naming**: Choose a unique bucket name.

**2. Attach a Bucket Policy**

Grant Elastic Load Balancing (ELB) permission to write logs to your S3 bucket by adding a bucket policy.

Here's an example policy for the Asia Pacific (Hyderabad) Region

json

CopyEdit

{

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

"Statement": [

{

"Effect": "Allow",

"Principal": {

"AWS": "arn:aws:iam::127311923021:root"

},

"Action": "s3:PutObject",

"Resource": "arn:aws:s3:::your-bucket-name/prefix/AWSLogs/your-account-id/\*"

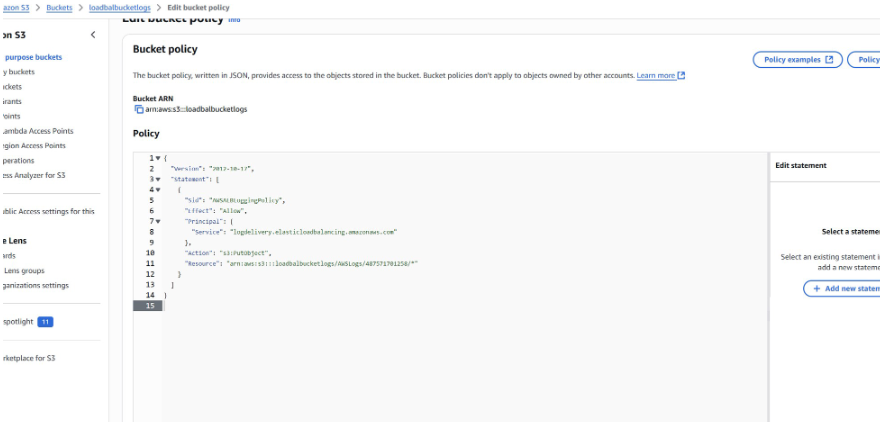
}

]

}

**Replace:**

* your-bucket-name with your S3 bucket name.
* prefix with your desired log prefix (optional).
* your-account-id with your AWS account ID.



**3. Enable Access Logs on Your ALB**

**Using the AWS Console:**

1. Navigate to the EC2 service.
2. In the left pane, select **Load Balancers**.
3. Choose your ALB and go to the **Attributes** tab.
4. Click **Edit attributes**.
5. Enable **Access logs**.
6. Specify the S3 URI in the format:
   * With prefix: s3://your-bucket-name/prefix
   * Without prefix: s3://your-bucket-name
7. Save the changes.

