

## Lesson 03 Demo 07

### Configuring an Application Load Balancer

**Objective:** To configure an Application Load Balancer in AWS to distribute traffic across multiple EC2 instances for load balancing and redundancy

**Tools required:** AWS Management Console, AWS EC2, and web browser

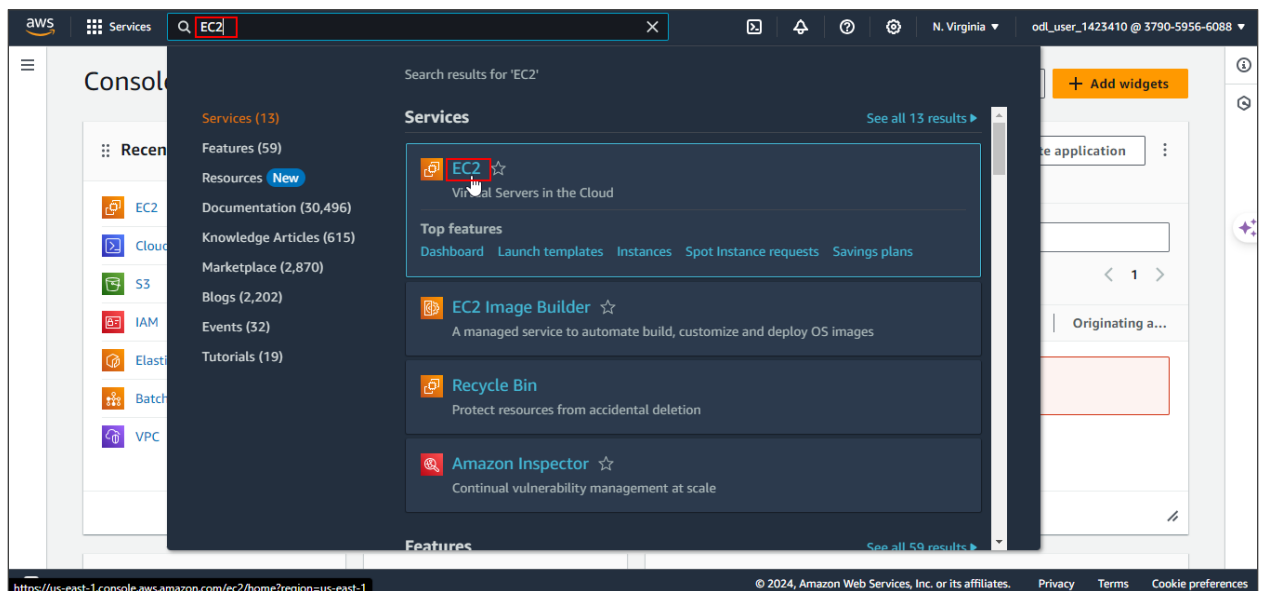
**Prerequisites:** None

Steps to be followed:

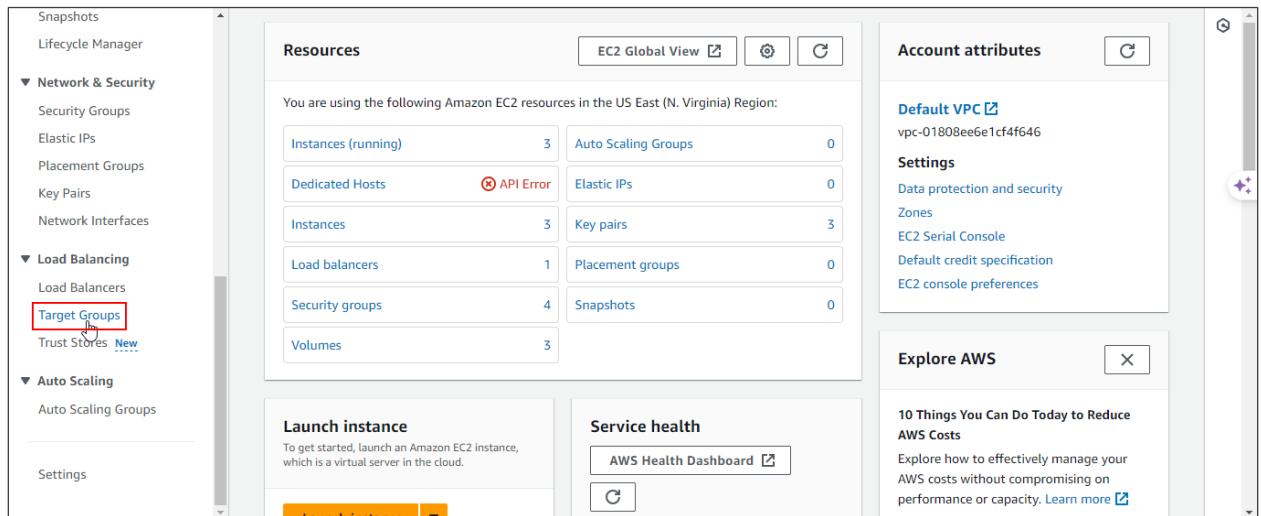
1. Create a target group
2. Launch EC2 instances
3. Configure the target group
4. Create a Load Balancer
5. Test the Load Balancer

#### Step 1: Create a target group

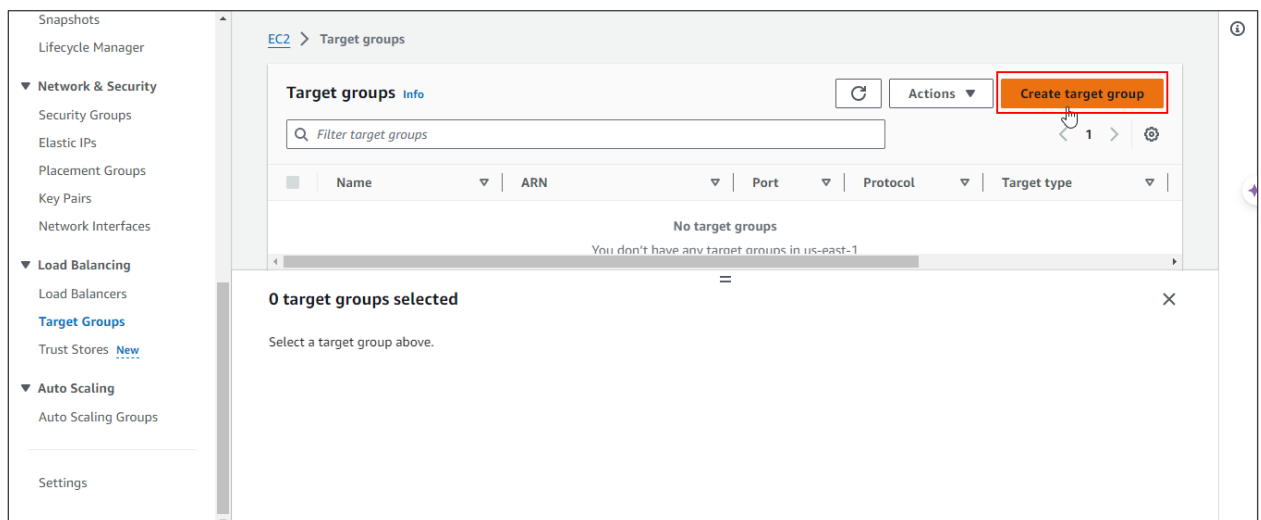
1.1 Navigate to the AWS console home dashboard, search for and click on **EC2**



## 1.2 Navigate to the Load Balancing section and click on Target Groups



## 1.3 Click on Create target group



## 1.4 In the **Basic configuration** section, choose **Instances** as the target type and enter a name for the target group, such as **MyTargetGroup**

EC2 > Target groups > Create target group

Step 1  
**Specify group details**

Step 2  
Register targets

### Specify group details

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

#### Basic configuration

Settings in this section can't be changed after the target group is created.

Choose a target type

- ☒ **Instances**
  - Supports load balancing to instances within a specific VPC.
  - Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.
- ☐ **IP addresses**
  - Supports load balancing to VPC and on-premises resources.
  - Facilitates routing to multiple IP addresses and network interfaces on the same instance.
  - Offers flexibility with microservice based architectures, simplifying inter-application communication.
  - Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Target group name

**MyTargetGroup**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Protocol : Port**

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.

HTTP 80

1-65535

**IP address type**

Only targets with the indicated IP address type can be registered to this target group.

- ☒ **IPv4**

Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.
- ☐ **IPv6**

Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#)

**VPC**

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

vpc-01808ee6e1cf4f646  
IPv4 VPC CIDR: 172.31.0.0/16

## 1.5 Set the protocol to **HTTP** and the path to **/index.html** in the **Health checks** section

Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

☐ gRPC  
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

### Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

**Health check protocol**  
HTTP

**Health check path**  
Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.  
/index.html  
Up to 1024 characters allowed.

► Advanced health check settings

### Attributes

## 1.6 Click on **Next**

**Health check path**  
Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.  
/index.html  
Up to 1024 characters allowed.

► Advanced health check settings

### Attributes

ⓘ Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

► **Tags - optional**  
Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

Cancel **Next**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 1.7 Review the configurations and click on **Create target group**

Port for routing traffic to the selected instances:  
1-65535 (separate multiple ports with commas)

80

Include as pending below

### Review targets

Targets (0) Remove all pending

All  < 1 > ⚙️

Remove	Health status	Instance ID	Name	Port	State	Security groups	Zone	Sub
No instances added yet								
Specify instances above, or leave the group empty if you prefer to add targets later.								

0 pending

Cancel Previous **Create target group**

Successfully created the target group: **MyTargetGroup**. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the **Targets** tab.

EC2 > Target groups > MyTargetGroup

## MyTargetGroup

Actions

### Details

arn:aws:elasticloadbalancing:us-east-1:379059566088:targetgroup/MyTargetGroup/ab5f6a4a1fe4668f

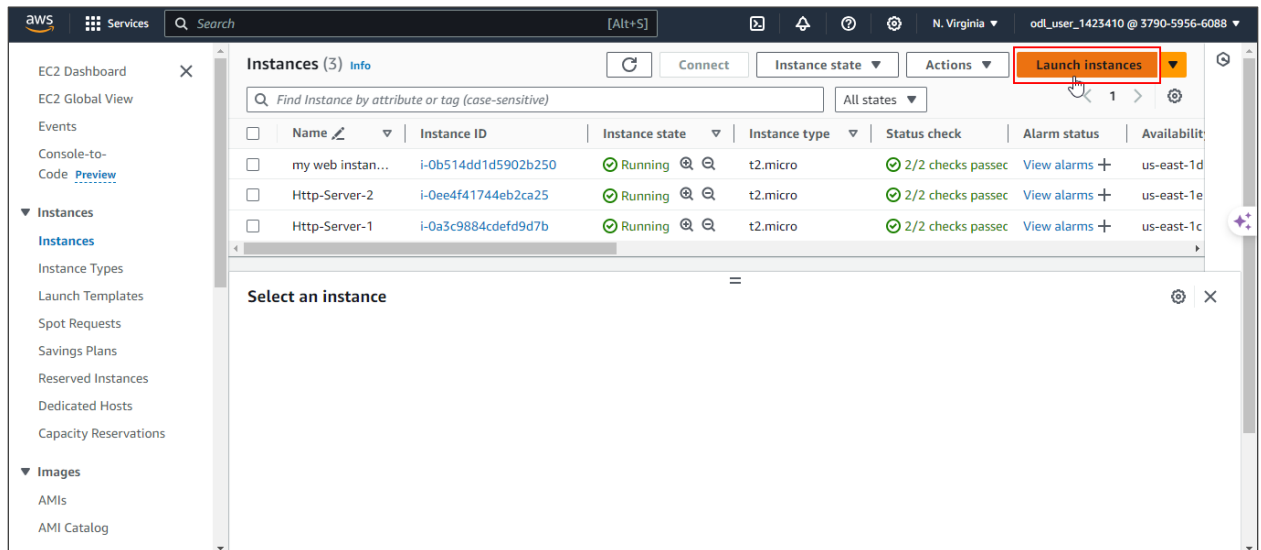
Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC <a href="#">vpc-01808ee6e1cf4f646</a>
IP address type IPv4	Load balancer <a href="#">None associated</a>		

0 Total targets	0 Healthy	0 Unhealthy	0 Unused	0 Initial	0 Draining
	0 Anomalous				

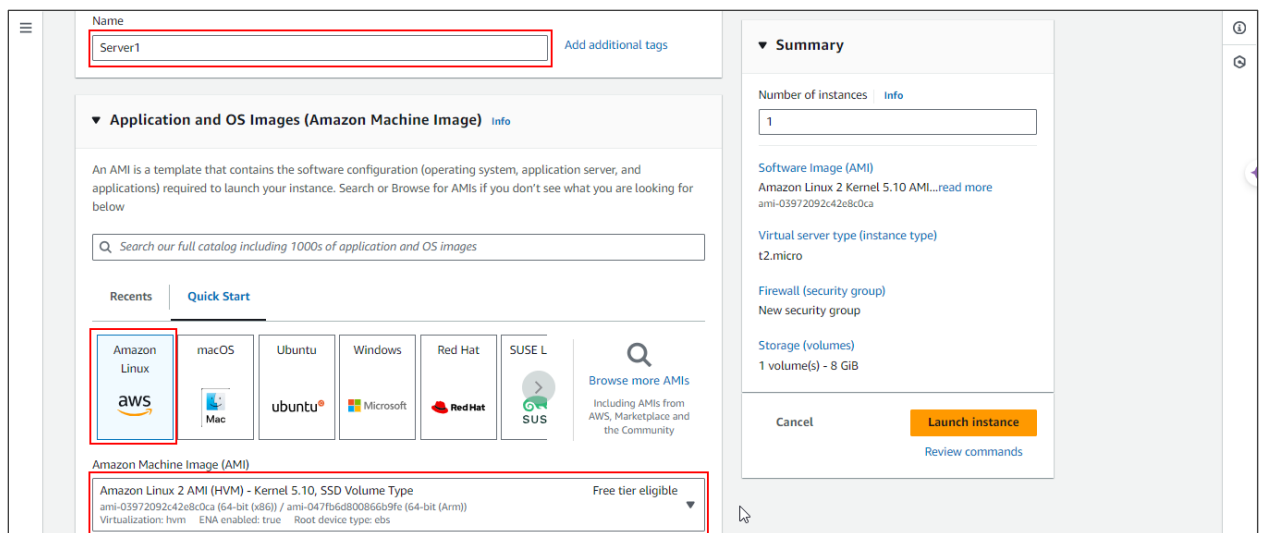
The target group has been successfully created.

## Step 2: Launch EC2 instances

### 2.1 Navigate to the Instances section and click on Launch instances



### 2.2 Provide a name for the instance and choose an appropriate AMI (Amazon Linux 2)



## 2.3 Select the instance type as **t2.micro**, create a key pair, and name it **Server-1**

**▼ Instance type** [Info](#) | [Get advice](#)

Instance type

t2.micro Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand RHEL base pricing: 0.026 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

Additional costs apply for AMIs with pre-installed software

☒ All generations [Compare instance types](#)

**▼ Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Server-1 [Create new key pair](#)

**▼ Network settings** [Info](#) [Edit](#)

**▼ Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[read more](#)

ami-03972092c42e8c0ca

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel [Launch instance](#) [Review commands](#)

## 2.4 Configure the network settings as shown:

**▼ Network settings** [Info](#)

VPC - required [Info](#)

vpc-01808ee6e1cf4f646 (default) [Refresh](#)

Subnet [Info](#)

No preference [Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

Security group name - required

launch-wizard-2

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-./!#,%&()\*+=&:[]\$\*

Description - required [Info](#)

launch-wizard-2 created 2024-08-12T13:50:25.331Z

**▼ Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[read more](#)

ami-03972092c42e8c0ca

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel [Launch instance](#) [Review commands](#)

**Inbound Security Group Rules**

Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

Type: [Info](#) | ssh | Protocol: [Info](#) | TCP | Port range: [Info](#) | 22

Source type: [Info](#) | Anywhere | Source: [Info](#) | Add CIDR, prefix list or security | 0.0.0.0/0 | Description - optional: [Info](#) | e.g. SSH for admin desktop

Security group rule 2 (TCP, 80, 0.0.0.0/0) Remove

Type: [Info](#) | HTTP | Protocol: [Info](#) | TCP | Port range: [Info](#) | 80

Source type: [Info](#) | Anywhere | Source: [Info](#) | Add CIDR, prefix list or security | 0.0.0.0/0 | Description - optional: [Info](#) | e.g. SSH for admin desktop

**Summary**

Number of instances: [Info](#) | 1

Software image (AMI) | Amazon Linux 2 Kernel 5.10 AMI...read more | ami-03972092c42e8c0ca

Virtual server type (instance type) | t2.micro

Firewall (security group) | New security group

Storage (volumes) | 1 volume(s) - 8 GiB

Cancel Launch instance [Review commands](#)

**Warning:** Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

2.5 Add the following user data script in the **Advance details** section, and click on **Launch instance**:

```
#!/bin/bash
yum update -y
yum install httpd -y
echo "<html><body><h1>This is Webserver1</h1></body></html>" >
/var/www/html/index.html
systemctl start httpd
systemctl enable httpd
```

**Advance details**

Allow tags in metadata: [Info](#) | Select

User data - optional: [Info](#) | Upload a file with your user data or enter it in the field. | Choose file

`#!/bin/bash
yum update -y
yum install httpd -y
echo "<html><body><h1>This is Webserver1</h1></body></html>" >
/var/www/html/index.html
systemctl start httpd
systemctl enable httpd`

☐ User data has already been base64 encoded

**Summary**

Number of instances: [Info](#) | 1

Software image (AMI) | Amazon Linux 2 Kernel 5.10 AMI...read more | ami-03972092c42e8c0ca

Virtual server type (instance type) | t2.micro

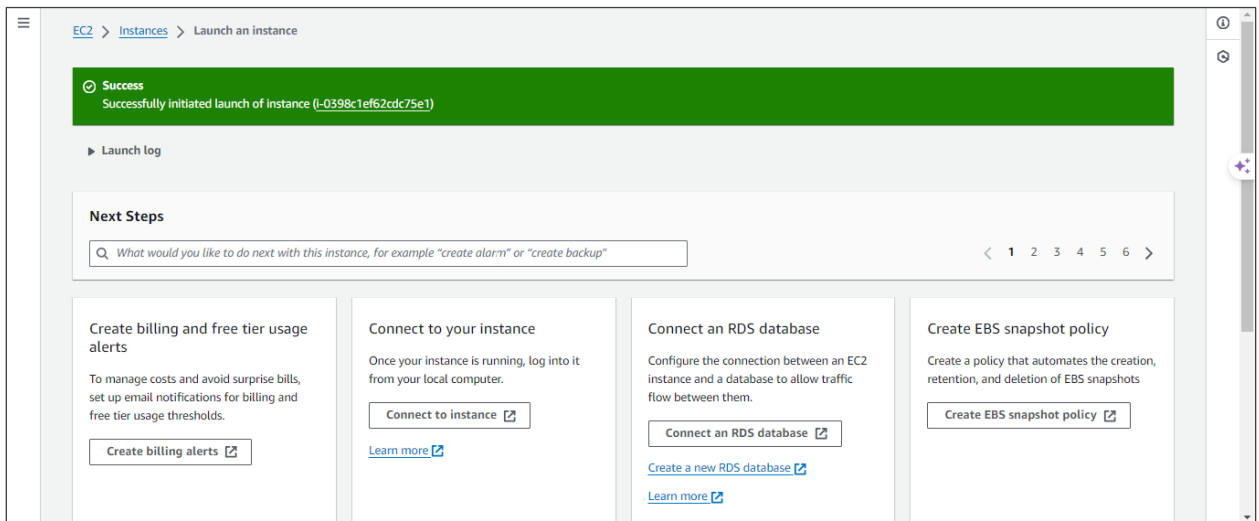
Firewall (security group) | New security group

Storage (volumes) | 1 volume(s) - 8 GiB

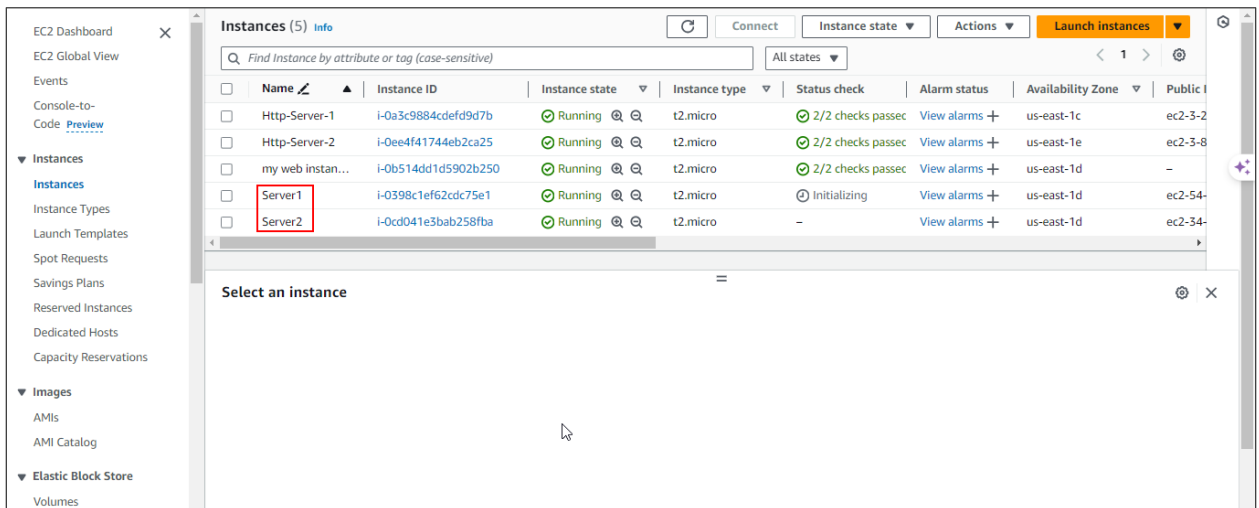
Cancel Launch instance [Review commands](#)



You will see the following interface:



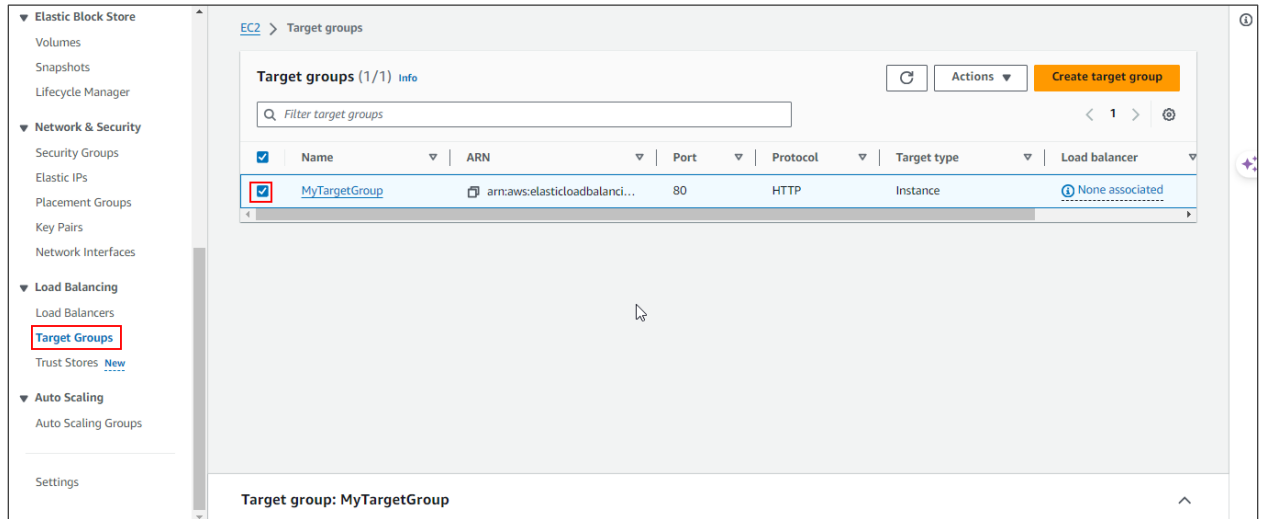
2.6 Launch another EC2 instance using the same steps, but modify the user data script to display the message **This is Webserver2**



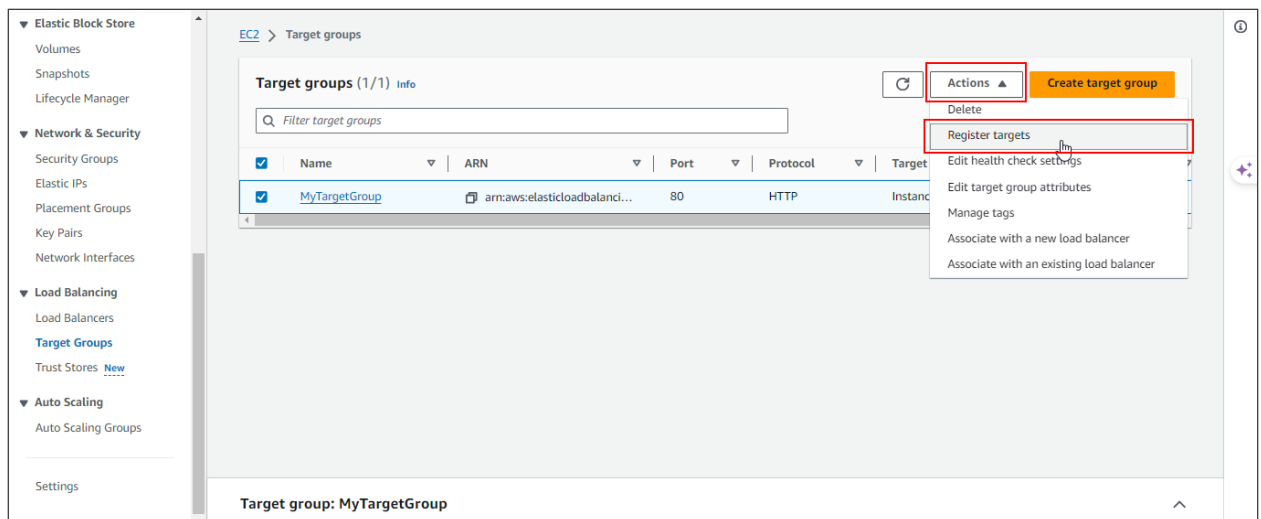
The EC2 instances have been successfully launched.

## Step 3: Configure the target group

### 3.1 Navigate to the **Target Groups** section and select the target group created in Step 1



### 3.2 Click on **Register targets** from the **Actions** menu



### 3.3 Select the instances (**Server1** and **Server2**) that were launched in Step 2 and click on **Include as pending below**

Available instances (2/4)

Filter instances

Instance ID	Name	State	Security groups	Zone
<input checked="" type="checkbox"/> i-0cd041e3bab258fba	Server2	Running	launch-wizard-3	us-east-1d
<input checked="" type="checkbox"/> i-0398c1ef62cdc75e1	Server1	Running	launch-wizard-2	us-east-1d
<input type="checkbox"/> i-0ee4f41744eb2ca25	Http-Server-2	Running	MyHttpServer	us-east-1e
<input type="checkbox"/> i-0a3c9884cdefd9d7b	Http-Server-1	Running	MyHttpServer	us-east-1c

2 selected

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**

### 3.4 Click on **Register pending targets** to register the instances with the target group

1-65535 (separate multiple ports with commas)

**Include as pending below**

2 selections are now pending below. Include more or register targets when ready.

**Review targets**

Targets (2)

Filter targets

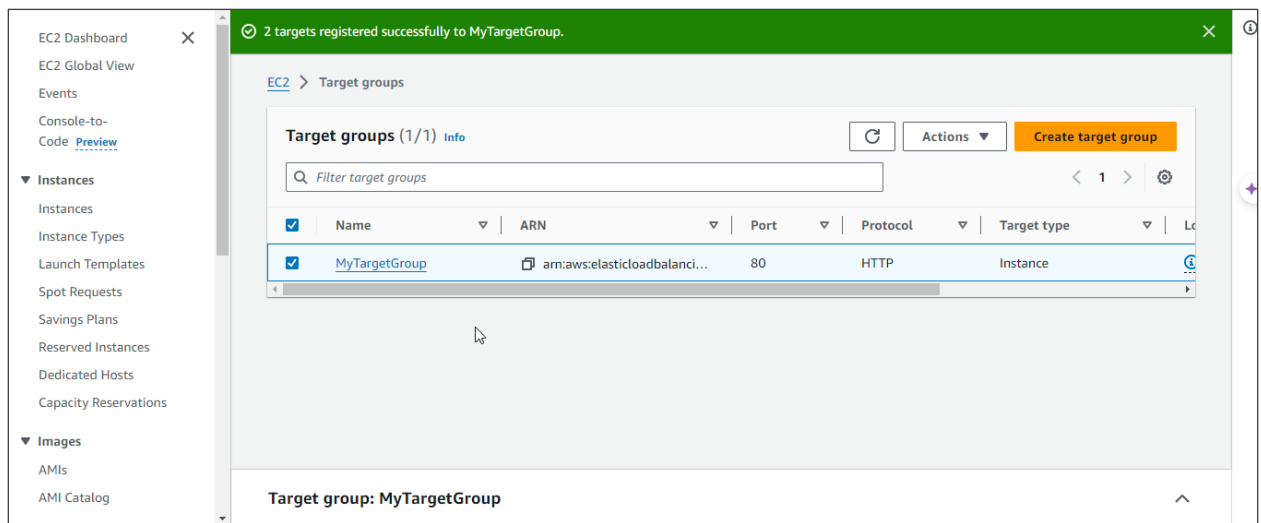
Show only pending

Remove all pending

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Laun
i-0cd041e3bab258fba	Server2	80	Running	launch-wizard-3	us-east-1d	172.31.89.225	subnet-005a925283ef33008	Augu
i-0398c1ef62cdc75e1	Server1	80	Running	launch-wizard-2	us-east-1d	172.31.95.124	subnet-005a925283ef33008	Augu

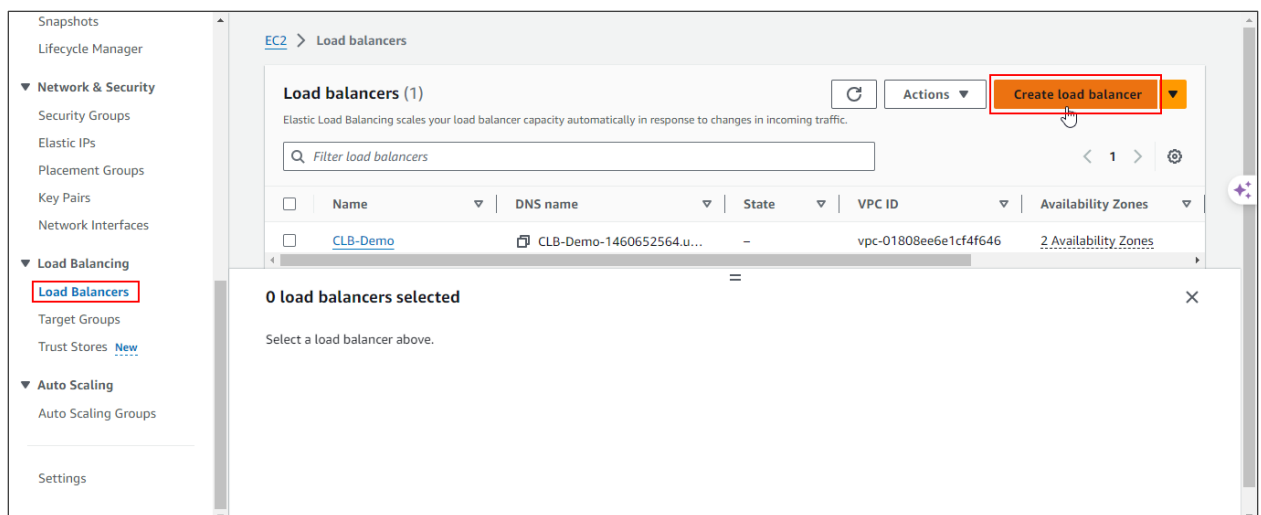
2 pending

Cancel **Register pending targets**



## Step 4: Create a Load Balancer

### 4.1 Navigate to the Load Balancers section under Load Balancing and click Create load balancer



## 4.2 Choose **Application Load Balancer** and click **Create**

**Application Load Balancer** [Info](#)

Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

**Create**

**Network Load Balancer** [Info](#)

Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

**Gateway Load Balancer** [Info](#)

Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

**Create**

CloudShell Feedback Create Application Load Balancer © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 4.3 Configure the load balancer settings, enter a name for the load balancer, such as **my-alb**, and select availability zones, such as **us-east-1a** and **us-east-1b**

**How Application Load Balancers work**

**Basic configuration**

**Load balancer name**  
Name must be unique within your AWS account and can't be changed after the load balancer is created.

my-alb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Scheme** [Info](#)  
Scheme can't be changed after the load balancer is created.

☒ **Internet-facing**  
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

☐ **Internal**  
An internal load balancer routes requests from clients to targets using private IP addresses. Compatible with the IPv4 and Dualstack IP address types.

**Load balancer IP address type** [Info](#)  
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

☒ **IPv4**  
Includes only IPv4 addresses.

☐ **Dualstack**  
Includes IPv4 and IPv6 addresses.

☐ **Dualstack without public IPv4**  
Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with Internet-facing load balancers only.

**Mappings** [Info](#)

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

**Availability Zones**

☒ **us-east-1a (use1-az4)**

Subnet

subnet-07ed5daff56349b53  
IPv4 subnet CIDR: 172.31.16.0/20

IPv4 address  
Assigned by AWS

☒ **us-east-1b (use1-az6)**

Subnet

subnet-08742ed07763637c1  
IPv4 subnet CIDR: 172.31.32.0/20

IPv4 address  
Assigned by AWS

☐ us-east-1c (use1-az1)

☐ us-east-1d (use1-az2)

☐ us-east-1e (use1-az3)

☐ us-east-1f (use1-az5)

4.4 Choose the default action for the listener configuration to accept HTTP traffic on port **80**, and select the target group created in Step 1

**Listeners and routing** [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80 [Remove](#)

Protocol: HTTP Port: 80

Default action: [Info](#)

Forward to: MyTargetGroup [Create target group](#) HTTP

Target type: Instance, IPv4

**Listener tags - optional**

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

[Add listener tag](#)

You can add up to 50 more tags.

[Add listener](#)

#### 4.5 Review the configuration and click **Create load balancer**

• IPv4

• us-east-1b  
subnet-08742ed07763637c1

Service integrations [Edit](#)

AWS WAF: None  
AWS Global Accelerator: None

Tags [Edit](#)

None

Attributes

ⓘ Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Creation workflow and status

► Server-side tasks and status

After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

Cancel **Create load balancer**

Wait until the **Status** changes from **Provisioning** to **Active**

EC2 Dashboard ×

EC2 Global View

Events

Console-to-Code [Preview](#)

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

▼ Images

AMIs

AMI Catalog

EC2 > Load balancers > my-alb

**my-alb**

⌂ Actions ▼

▼ Details

Load balancer type Application	Status 🟢 Active	VPC <a href="#">vpc-01808ee6e1cf4f646</a>	Load balancer IP address type IPv4
Scheme Internet-facing	Hosted zone Z35SXDOTRQ7X7K	Availability Zones <a href="#">subnet-08742ed07763637c1</a> us-east-1b (use1-az6) <a href="#">subnet-07ed5daff56349b53</a> us-east-1a (use1-az4)	Date created August 12, 2024, 19:51 (UTC+05:30)

Load balancer ARN  
arn:aws:elasticloadbalancing:us-east-1:379059566088:loadbalancer/app/my-alb/0bdb80cffc42d9e7

DNS name [Info](#)  
my-alb-505116050.us-east-1.elb.amazonaws.com (A Record)

[Listeners and rules](#) | [Network mapping](#) | [Resource map - new](#) | [Security](#) | [Monitoring](#) | [Integrations](#) | [Attributes](#) | [Tags](#)

#### 4.6 Click on the **Security** tab

The screenshot shows the AWS Management Console interface for a load balancer named 'my-alb'. The left sidebar contains navigation options like 'EC2 Dashboard', 'Instances', and 'Images'. The main content area displays the 'Details' tab for the load balancer, showing its status as 'Active', VPC ID, Availability Zones, and Load balancer ARN. The bottom navigation bar has several tabs: 'Listeners and rules', 'Network mapping', 'Resource map - new', 'Security' (which is highlighted with a red box and a mouse cursor), 'Monitoring', 'Integrations', 'Attributes', and 'Tags'.

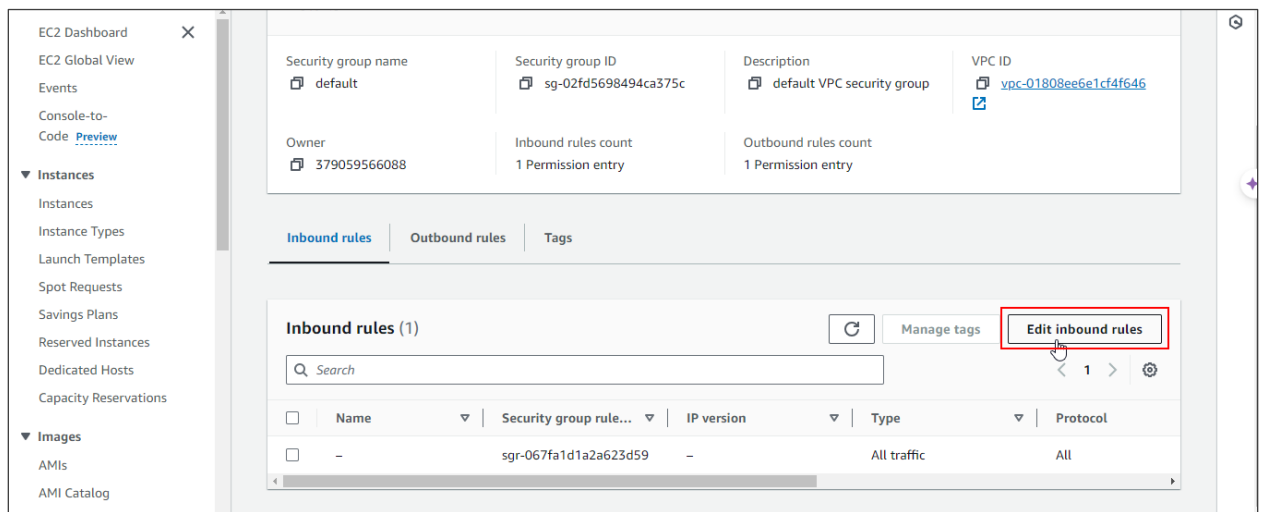
#### 4.7 Click on the **Security Group ID** name

This screenshot shows the 'Security' tab for the 'my-alb' load balancer. It displays a table titled 'Security groups (1)' with one entry. The 'Security Group ID' 'sg-02fd5698494ca375c' is highlighted with a red box and a mouse cursor. The table also shows the 'Name' as 'default' and the 'Description' as 'default VPC security group'. The top part of the console shows the load balancer's details, including its status and VPC configuration.

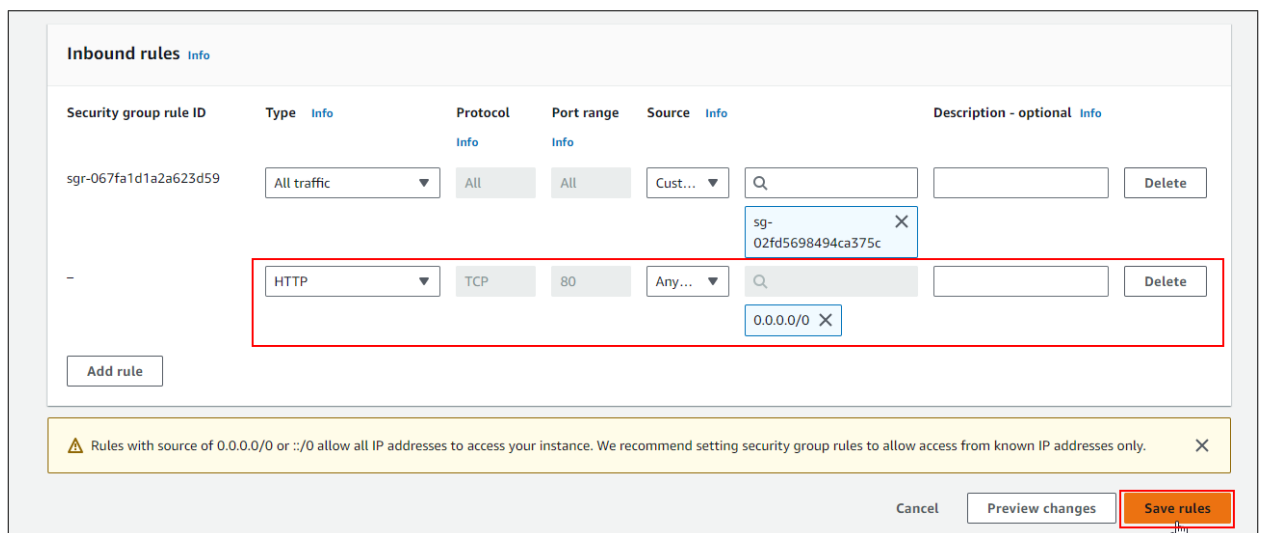
Security Group ID	Name	Description
sg-02fd5698494ca375c	default	default VPC security group



#### 4.8 Click on **Edit inbound rules**



#### 4.9 Create an inbound rule to permit port **80** access for all, and click on **Save rules** as shown:



## Step 5: Test the Load Balancer

### 5.1 Navigate to the **Target Groups** section and select the target group you created

The screenshot shows the AWS Management Console interface for the 'Target groups' section. On the left, there is a navigation menu with categories like 'Elastic Block Store', 'Network & Security', 'Load Balancing', and 'Auto Scaling'. The 'Load Balancing' section is expanded, and 'Target Groups' is selected. The main content area shows a list of target groups with a table containing columns: Name, ARN, Port, Protocol, Target type, and Load balancer. One target group, 'MyTargetGroup', is selected. Below the table, the 'Details' tab is active, showing the ARN and a table with the following information:

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-014d96d5e406276f2

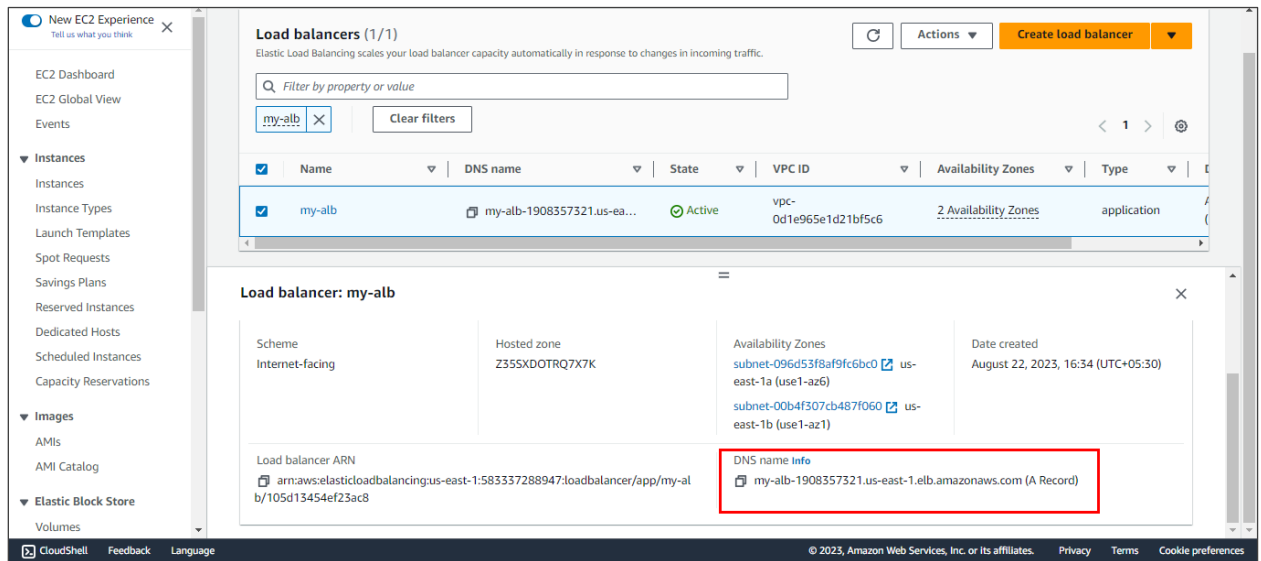
### 5.2 Click on **Details** to verify that your instances are registered and healthy

The screenshot shows the 'Details' tab for the 'MyTargetGroup' target group. It displays the IP address type as 'IPv4' and the load balancer as 'my-alb'. Below this, there is a table showing the status of the targets:

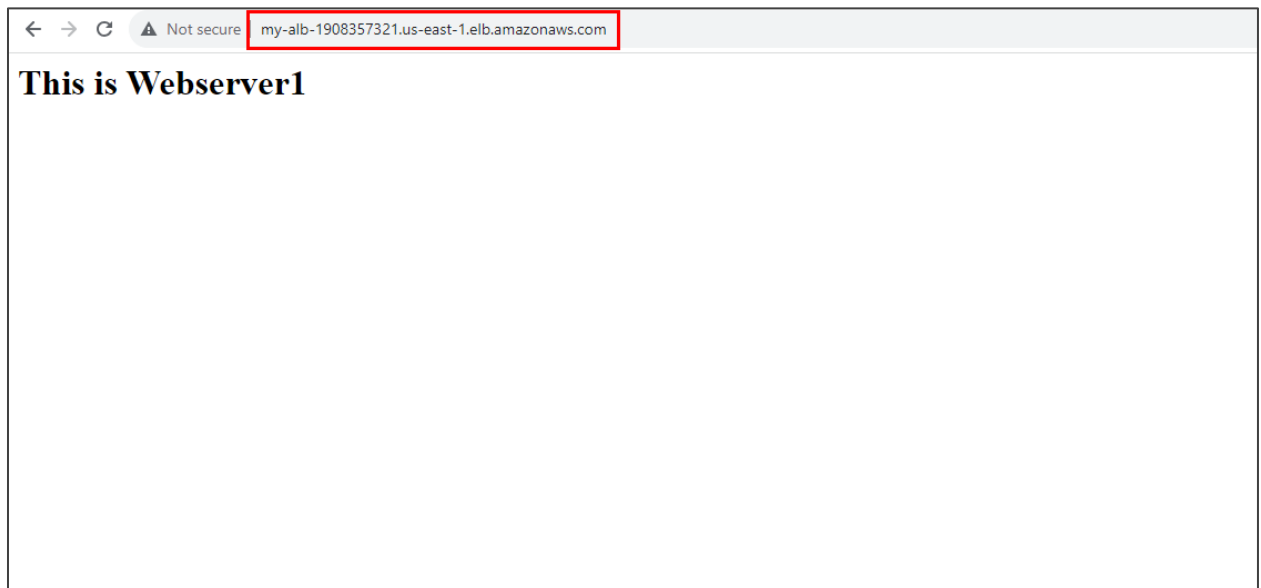
Total targets	Healthy	Unhealthy	Unused	Initial	Draining
2	2	0	0	0	0

The 'Healthy' status is highlighted with a red box. Below the table, there is a section titled 'Distribution of targets by Availability Zone (AZ)' with a note: 'Select values in this table to see corresponding filters applied to the Registered targets table below.'

### 5.3 Navigate to the **Load Balancers** section and copy the DNS name of the Load Balancer

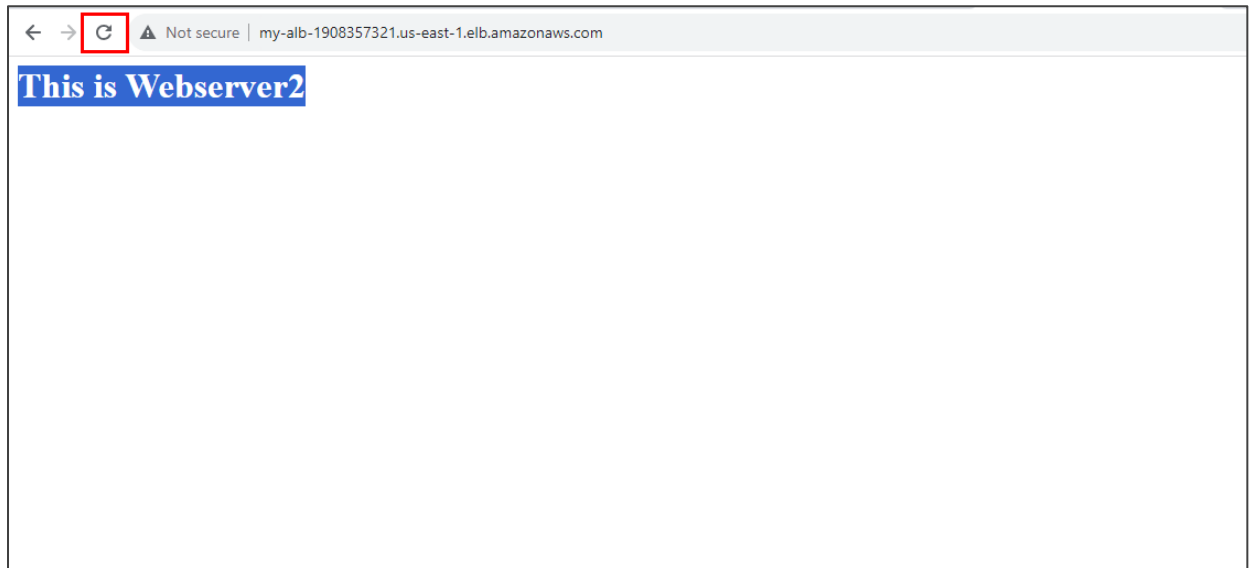


### 5.4 Open a browser window and paste the DNS URL into the address bar



You will observe the header message originating from the **Server1** instance.

5.5 Refresh the web page multiple times to see the header message originating from the **Server2** instance



By following these steps, you have successfully configured an Application Load Balancer in AWS to distribute traffic across multiple EC2 instances, ensuring load balancing and redundancy.