Project in AWS
Practice Lab

Use Application Load Balancers for Web Servers

Andra-Diana Popescu

2025

ABOUT THIS LAB

Load balancing automatically distributes your incoming traffic across multiple targets, such as EC2 instances, containers, and IP addresses, in one or more Availability Zones. In this lab, we configure an Application Load Balancer to distribute network traffic to two EC2 instances. We then enable stickiness, so that once a server is contacted, the user is always sent to that server. This ensures our legacy application continues to work despite not supporting distributed logins. By the end of this lab, the user will understand how to create an Application Load Balancer and enable sticky sessions.

LEARNING OBJECTIVES

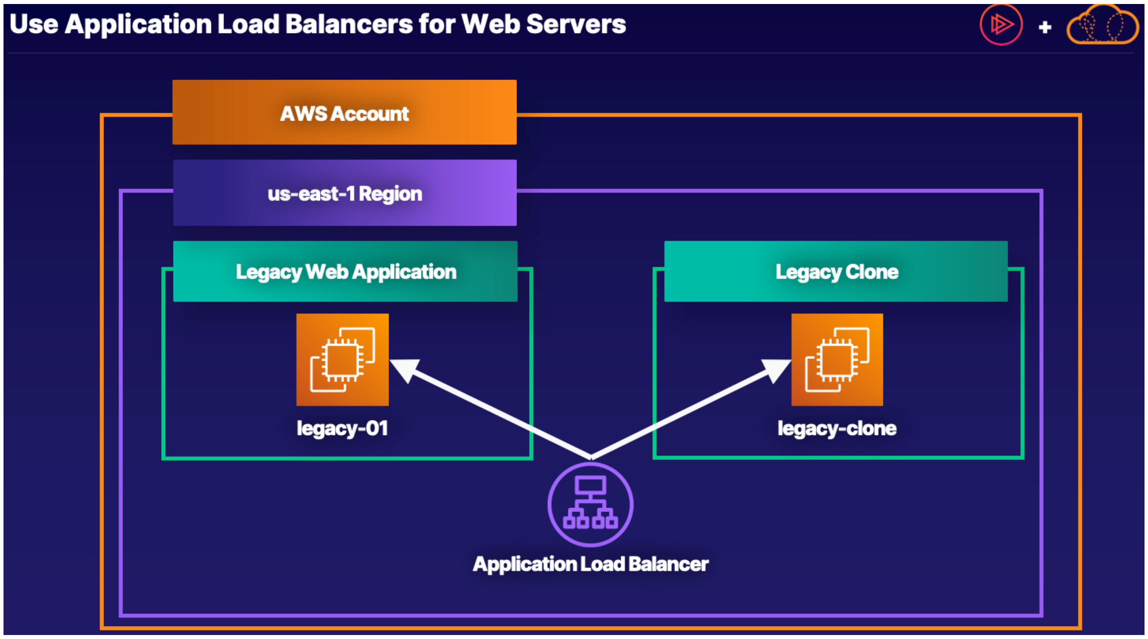
- Observe the Provided EC2 Website and Create a Second Server
- Create an Application Load Balancer
- Enable Sticky Sessions

AWS Documentation about ALB: https://aws.amazon.com/elasticloadbalancing/application-load-balancer/#topic-0

Source: https://learn.acloud.guru/course/certified-solutions-architect-associate/

Table of Contents

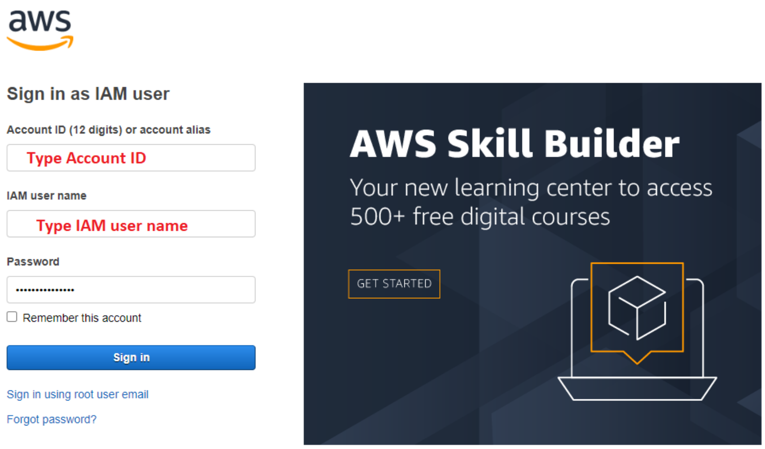
Lab Diagram



We have the AWS account in **us-east-1** Region, and we have an EC2 instance. Our scenario is that your company has a legacy web application that needs to be scaled up to run on multiple web servers. The application is very old and highly stateful, and doesn't support logins across multiple servers.

To scale the application, but to still ensure users continue to use the same server each time they visit the website, we'll set up a clone of our legacy website, and then we'll create an application load balancer with sticky sessions to manage the connections. This will prevent a user from accessing a different web server if they visit more than once, thus keeping this highly stateful website working correctly.

Log in to your AWS account



Sign in as IAM user

Account ID (12 digits) or account alias

Type Account ID

IAM user name

Type IAM user name

Password

················

☐ Remember this account

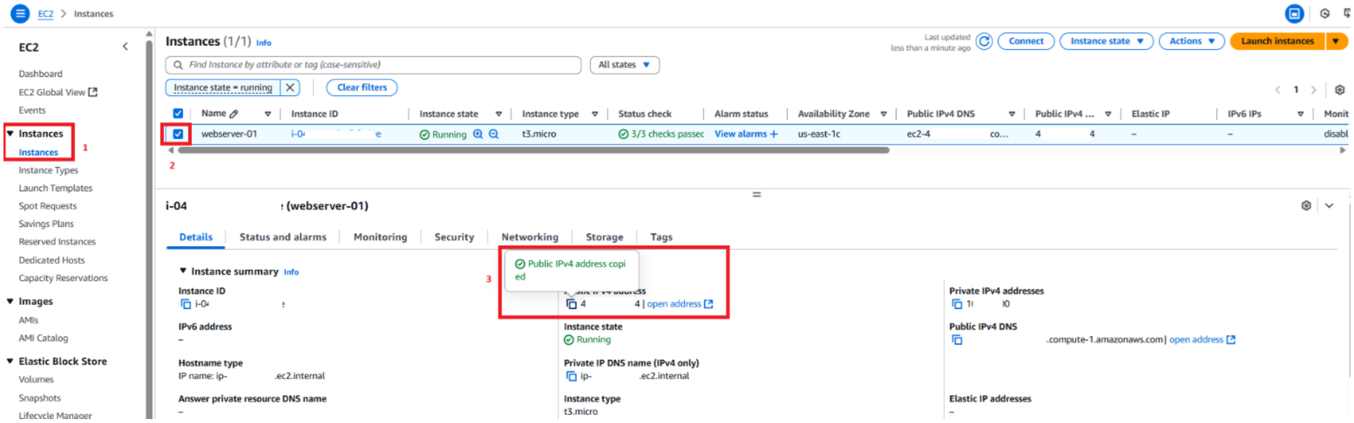
Sign in
Sign in using root user email
Forgot password?
AWS Skill Builder
Your new learning center to access 500+ free digital courses
GET STARTED

1. Observe the Provided EC2 Website and Create a Second Server

1. Once you are logged in to the AWS Management Console, navigate to **EC2 → Instances**.

2. Click the checkbox next to ***webserver-01***. The instance details display below.

3. Copy its Public IPv4 address. Do NOT try clicking on the *open address* link as it won't work.



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monit
☑	webserver-01	i-0· e	Running	t3.micro	3/3 checks passec	View alarms +	us-east-1c	ec2-4 co...	4 4	–	–	disabl

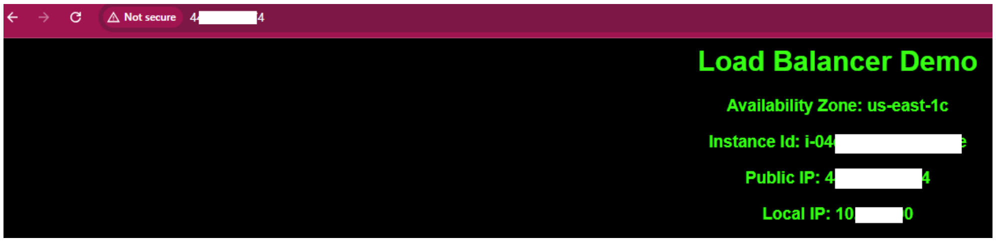
i-04 : (webserver-01)

Instance summary Info

Public IPv4 address copied

Instance ID i-0· :	Public IPv4 address 4 4 | open address	Private IPv4 addresses 1( )0
IPv6 address –	Instance state Running	Public IPv4 DNS .compute-1.amazonaws.com| open address
Hostname type IP name: ip- .ec2.internal	Private IP DNS name (IPv4 only) ip- .ec2.internal	
Answer private resource DNS name –	Instance type t3.micro	Elastic IP addresses –

4. In a new browser tab, paste in the public IP address you just copied. You should see the load balancer demo page. This is how we're going to identify which instance we end up on, once we have the load balancer set up.



Load Balancer Demo

Availability Zone: us-east-1c

Instance Id: i-04[redacted]

Public IP: 4[redacted]4

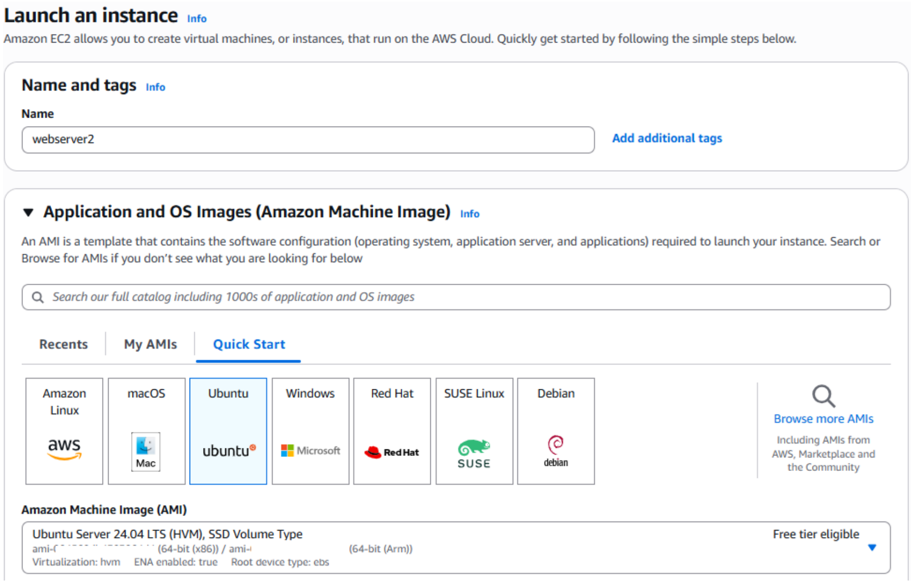
Local IP: 10[redacted]0

5. Now, let’s create another EC2 instance.

6. Back in the EC2 console, at the top, click **Launch instances**.

7. Under ***Name and Tags***, enter "***webserver2***".

8. Under ***Application and OS Images (Amazon Machine Image)***, select **Ubuntu and Ubuntu Server 24.04 LTS**.



Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: webserver2

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don’t see what you are looking for below

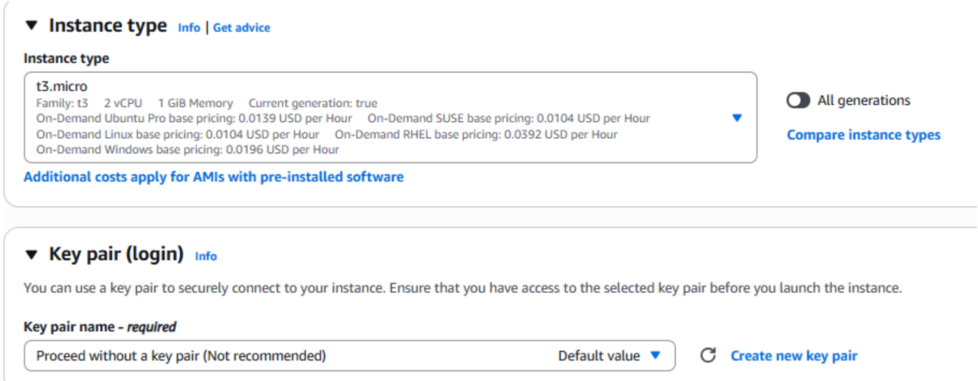
Amazon Machine Image (AMI): Ubuntu Server 24.04 LTS (HVM), SSD Volume Type — Free tier eligible

ami-[redacted] (64-bit (x86)) / ami-[redacted] (64-bit (Arm))

Virtualization: hvm   ENA enabled: true   Root device type: ebs

9. Under ***Instance Type***, select **t3.micro**.

10. Under ***Key pair (login)***, in the dropdown, select **Proceed without a key pair**.



Instance type Info | Get advice

Instance type: t3.micro

Family: t3   2 vCPU   1 GiB Memory   Current generation: true

On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour   On-Demand SUSE base pricing: 0.0104 USD per Hour

On-Demand Linux base pricing: 0.0104 USD per Hour   On-Demand RHEL base pricing: 0.0392 USD per Hour

On-Demand Windows base pricing: 0.0196 USD per Hour

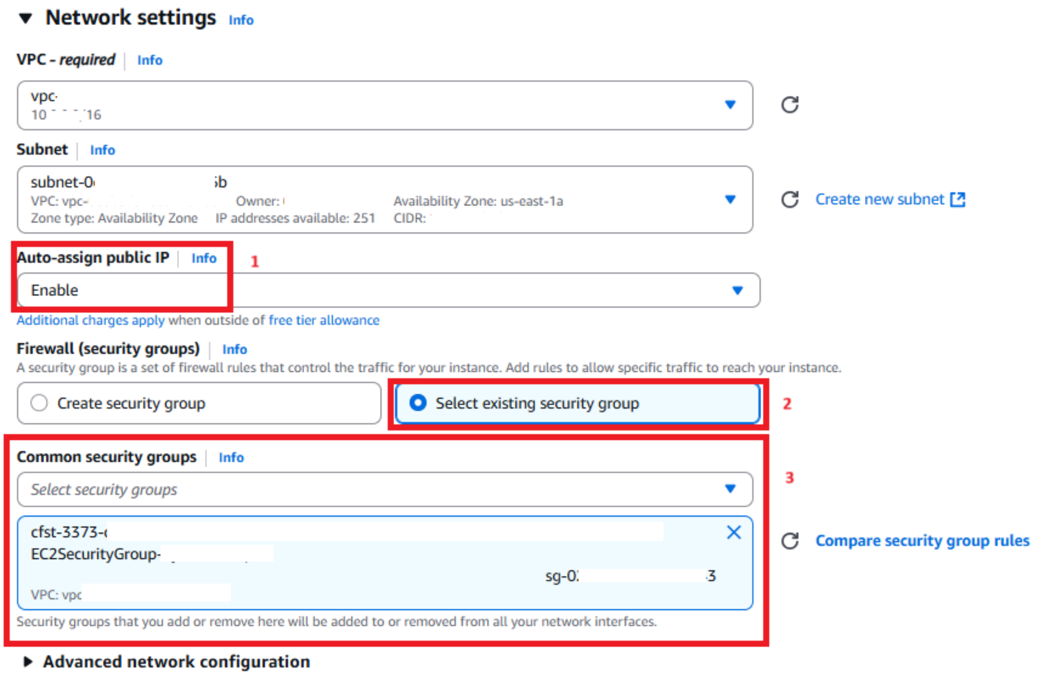
Key pair (login) Info

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*: Proceed without a key pair (Not recommended) — Default value

11. Under ***Network settings***, click **Edit** and set **Auto-assign Public IP** to **Enable**.

12. Under ***Network settings*** > ***Firewall (security groups)***, click **Select existing security group** and select the one with **EC2SecurityGroup** in its name (not the default security group).



▼ Network settings Info

VPC - *required* | Info

vpc-
10 16

Subnet | Info

subnet-0 ib
VPC: vpc- Owner: Availability Zone: us-east-1a
Zone type: Availability Zone IP addresses available: 251 CIDR:

Create new subnet

Auto-assign public IP | Info 1

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 2

Common security groups | Info 3

Select security groups

cfst-3373-
EC2SecurityGroup-
sg-0 3
VPC: vpc

Compare security group rules

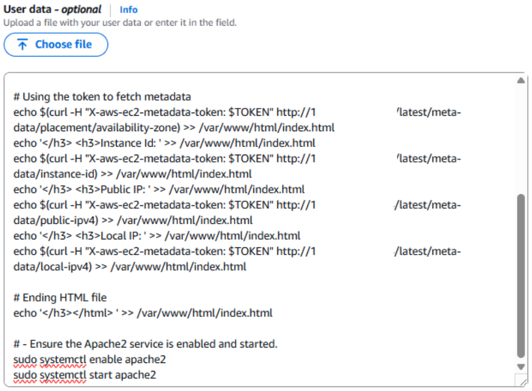
Security groups that you add or remove here will be added to or removed from all your network interfaces.

▶ Advanced network configuration

13. Under ***Advanced Details***, in the **User Data** box, enter the following bootstrap script:

```
#!/bin/bash

# Update and install necessary packages

sudo apt-get update -y

sudo apt-get install -y apache2 unzip

# Fetching the token for IMDSv2

TOKEN=`curl -X PUT "http://169.***.***.**4/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`

# Starting HTML file

echo '<html><center><body bgcolor="black" text="#39ff14" style="font-family: Arial"><h1>Load Balancer Demo</h1><h3>Availability Zone: ' > /var/www/html/index.html

# Using the token to fetch metadata

echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http:// 169.***.***.**4/latest/meta-data/placement/availability-zone) >> /var/www/html/index.html
echo '</h3> <h3>Instance Id: ' >> /var/www/html/index.html

echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http:// 169.***.***.**4/latest/meta-data/instance-id) >> /var/www/html/index.html echo '</h3> <h3>Public IP: ' >> /var/www/html/index.html

echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http:// 169.***.***.**4/latest/meta-data/public-ipv4) >> /var/www/html/index.html echo '</h3> <h3>Local IP: ' >> /var/www/html/index.html

echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http:// 169.***.***.**4/latest/meta-data/local-ipv4) >> /var/www/html/index.html

# Ending HTML file

echo '</h3></html> ' >> /var/www/html/index.html

# Ensure the Apache2 service is enabled and started.

sudo systemctl enable apache2

sudo systemctl start apache2
```

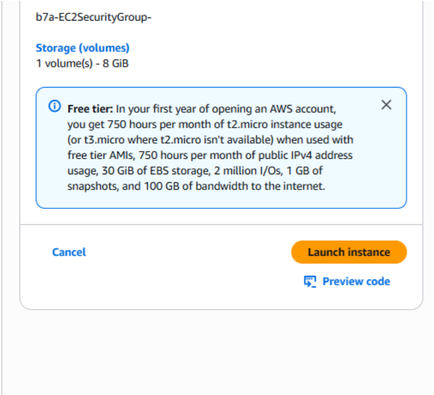
14. Click **Launch Instance**.



User data - *optional* | Info

Upload a file with your user data or enter it in the field.

```
# Using the token to fetch metadata
echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http://1                'latest/meta-
data/placement/availability-zone) >> /var/www/html/index.html
echo '</h3> <h3>Instance Id: ' >> /var/www/html/index.html
echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http://1                'latest/meta-
data/instance-id) >> /var/www/html/index.html
echo '</h3> <h3>Public IP: ' >> /var/www/html/index.html
echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http://1                /latest/meta-
data/public-ipv4) >> /var/www/html/index.html
echo '</h3> <h3>Local IP: ' >> /var/www/html/index.html
echo $(curl -H "X-aws-ec2-metadata-token: $TOKEN" http://1                /latest/meta-
data/local-ipv4) >> /var/www/html/index.html

# Ending HTML file
echo '</h3></html> ' >> /var/www/html/index.html

# - Ensure the Apache2 service is enabled and started.
sudo systemctl enable apache2
sudo systemctl start apache2
```



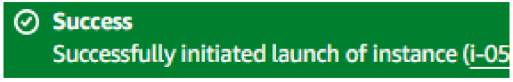
b7a-EC2SecurityGroup-

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

15. Click the **Instance ID** (this will start with i-).

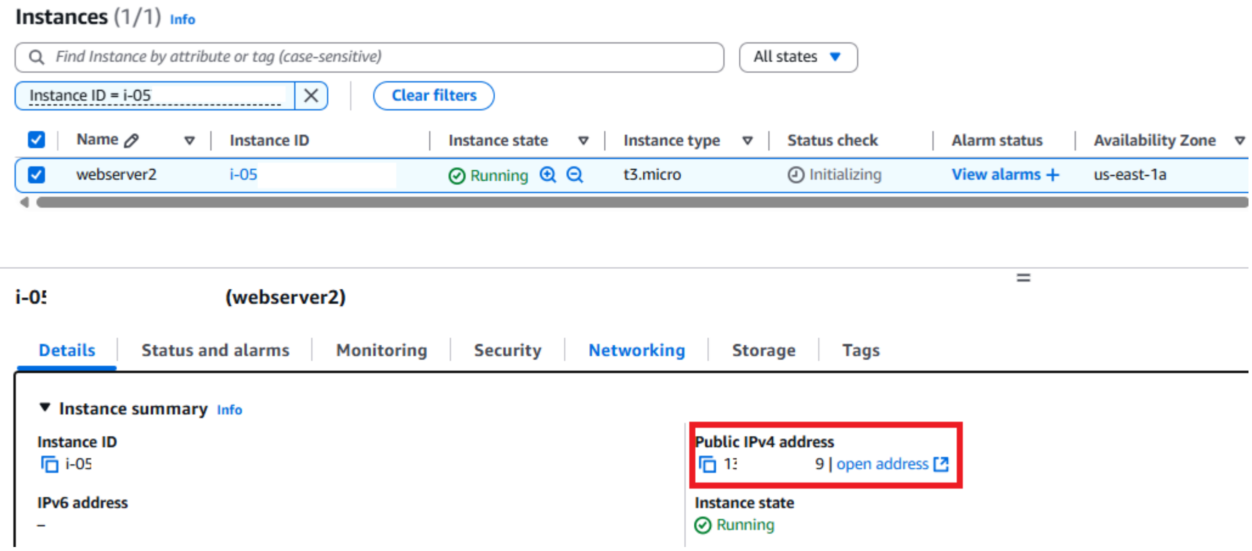


Success

Successfully initiated launch of instance (i-05

16. Once it's in the Running state, **copy the Public IPv4 address**.

Note: Do NOT try clicking on the open address link as it won't work.


Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive) | All states ▼

Instance ID = i-05 ✕ | Clear filters

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
☑	webserver2	i-05	Running	t3.micro	Initializing	View alarms +	us-east-1a

i-0! [redacted] (webserver2)

▼ Instance summary Info

Instance ID
i-05

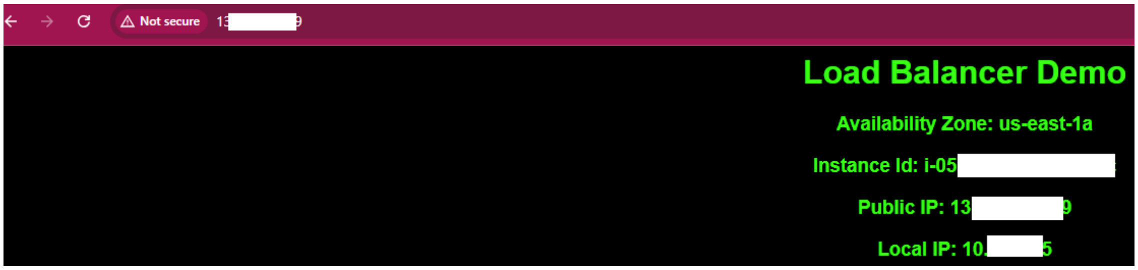
IPv6 address
–

Public IPv4 address
1: [redacted] 9 | open address

Instance state
Running

17. In a new browser tab, paste in the public IP address you just copied. You should see the load balancer demo page again, which means the legacy clone is successfully running. This time, though, it will have a different instance ID, public IP, and local IP listed.

    Note: If your second EC2 doesn't open the demo page, it may need a couple of minutes to finish provisioning. Wait for the Status check column to show "2/2 checks passed".



1 [redacted] 9

Load Balancer Demo

Availability Zone: us-east-1a

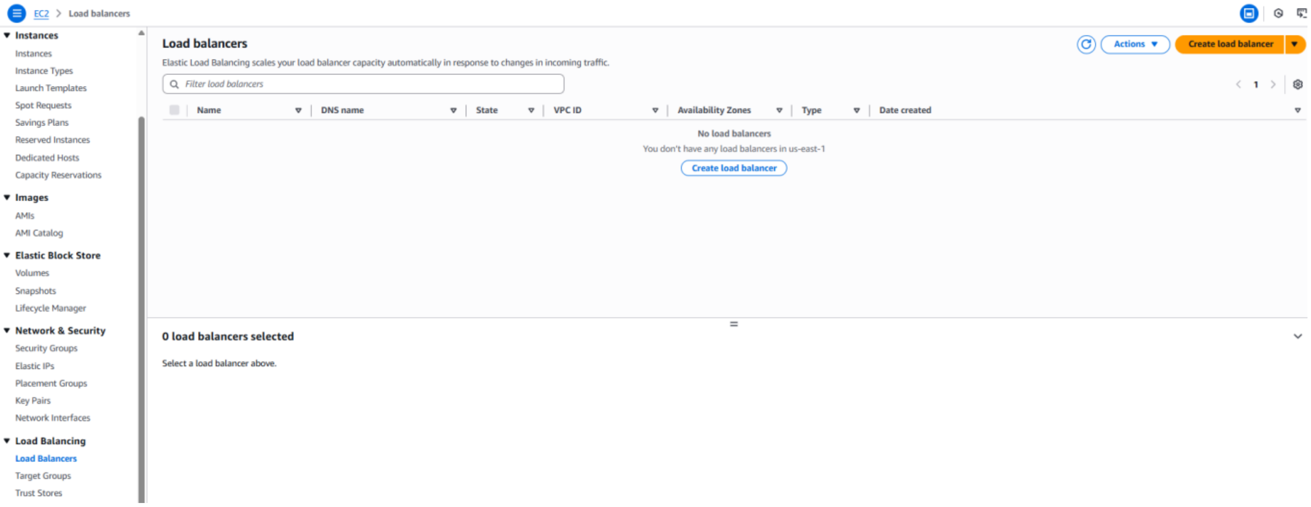
Instance Id: i-05 [redacted]

Public IP: 13 [redacted] 9

Local IP: 10. [redacted] 5

2. Create an Application Load Balancer

1. Back in the EC2 console, click **Load Balancers** in the left-hand menu.
2. Click **Create Load Balancer**.



Load balancers

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	DNS name	State	VPC ID	Availability Zones	Type	Date created

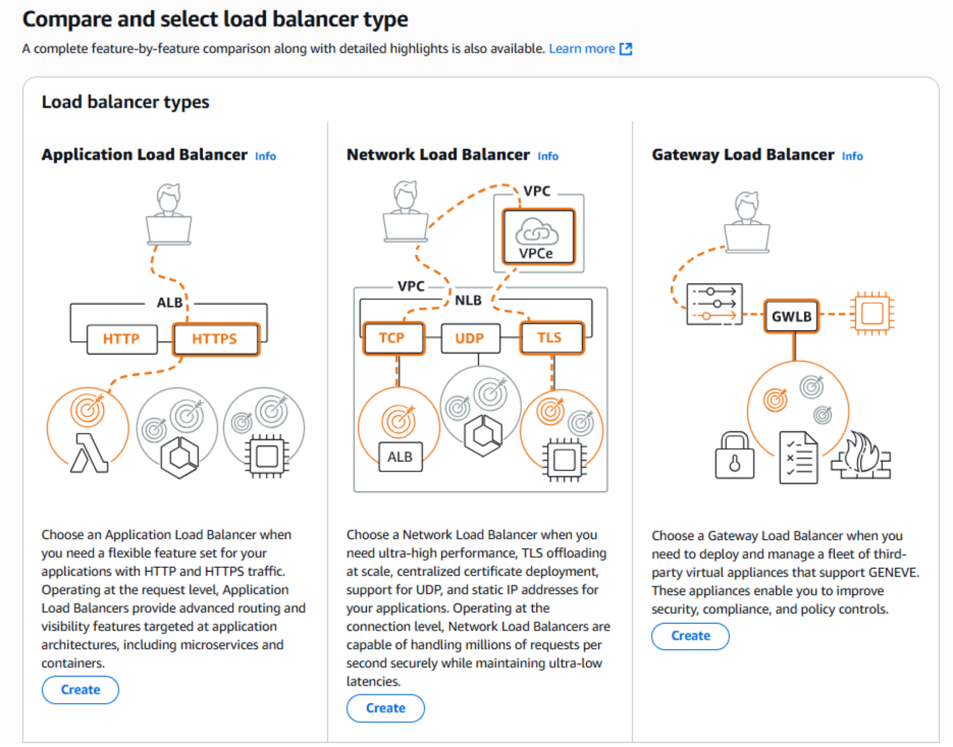
No load balancers

You don't have any load balancers in us-east-1

0 load balancers selected

Select a load balancer above.

3. From the ***Application Load Balancer*** card, click **Create**.



Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. Learn more

Load balancer types

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.

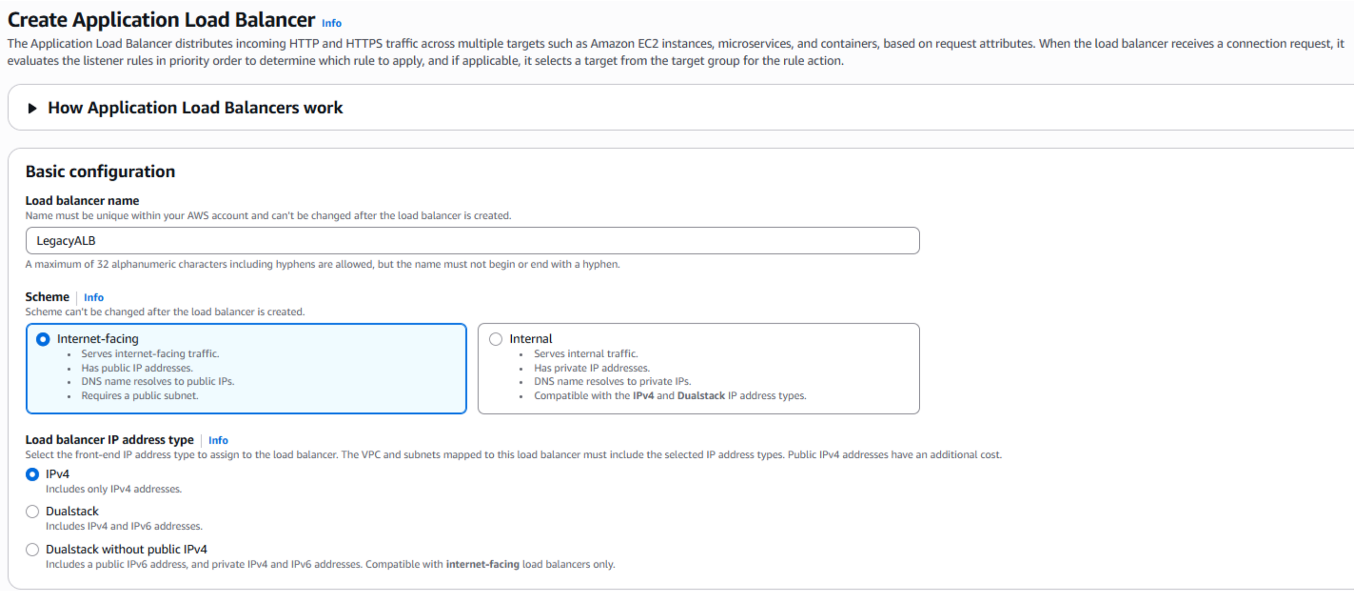
Network Load Balancer Info

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.

4. For ***Load balancer name***, enter "**LegacyALB**".



Create Application Load Balancer Info

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

▶ 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.

LegacyALB

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
- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

○ Internal
- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- 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.

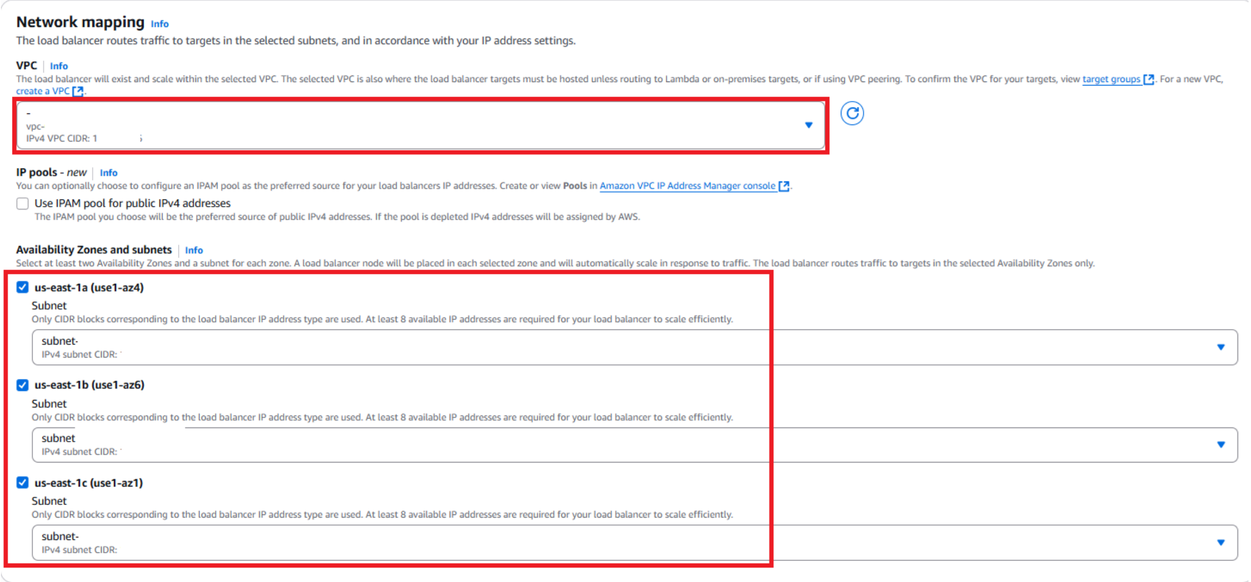
● IPv4
Includes only IPv4 addresses.

○ 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.

5. Under ***Network mapping***, click the **VPC** dropdown, and select the listed VPC.

6. When the ***Availability Zones*** list pops up, **select** each one (us-east-1a, us-east-1b, and us-east-1c).



Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC | Info
The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view target groups. For a new VPC, create a VPC.

-
vpc-
IPv4 VPC CIDR: 1

IP pools - *new* | Info
You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view **Pools** in Amazon VPC IP Address Manager console.

□ Use IPAM pool for public IPv4 addresses
The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

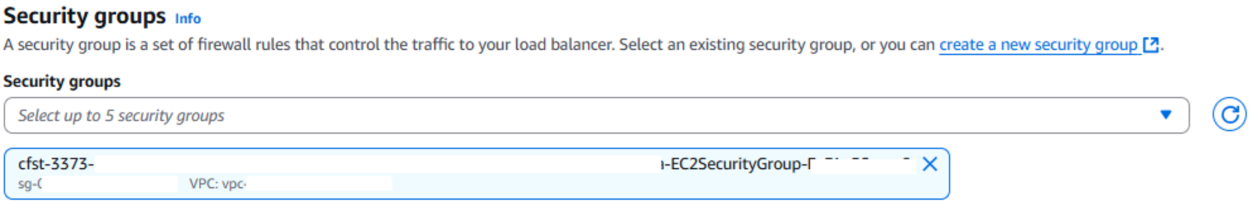
Availability Zones and subnets | Info
Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

☑ **us-east-1a (use1-az4)**
Subnet
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.
subnet-
IPv4 subnet CIDR:

☑ **us-east-1b (use1-az6)**
Subnet
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.
subnet
IPv4 subnet CIDR:

☑ **us-east-1c (use1-az1)**
Subnet
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.
subnet-
IPv4 subnet CIDR:

7. Under ***Security groups***, deselect the default security group listed, and select the one from the dropdown with EC2SecurityGroup in its name.



Security groups Info

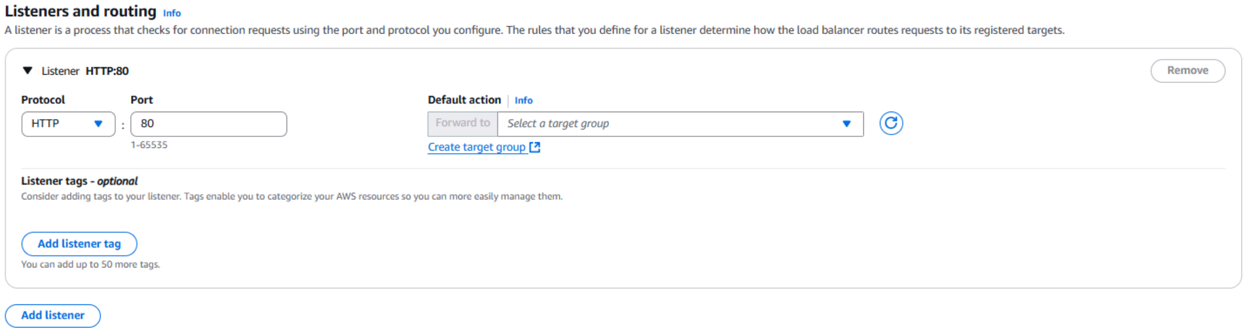
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group.

Security groups

Select up to 5 security groups

cfst-3373- ı-EC2SecurityGroup-Γ ✕
sg-(VPC: vpc-

8. Under ***Listeners and routing***, ensure that the **Protocol** is set to **HTTP** and the **Port** is **80**. Then, under ***Default action***, click **Create target group**. This will open a new tab. Keep this first tab open to complete later.



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 (1-65535)

Default action | Info

Forward to: *Select a target group* ▼

Create target group

Listener tags - *optional*

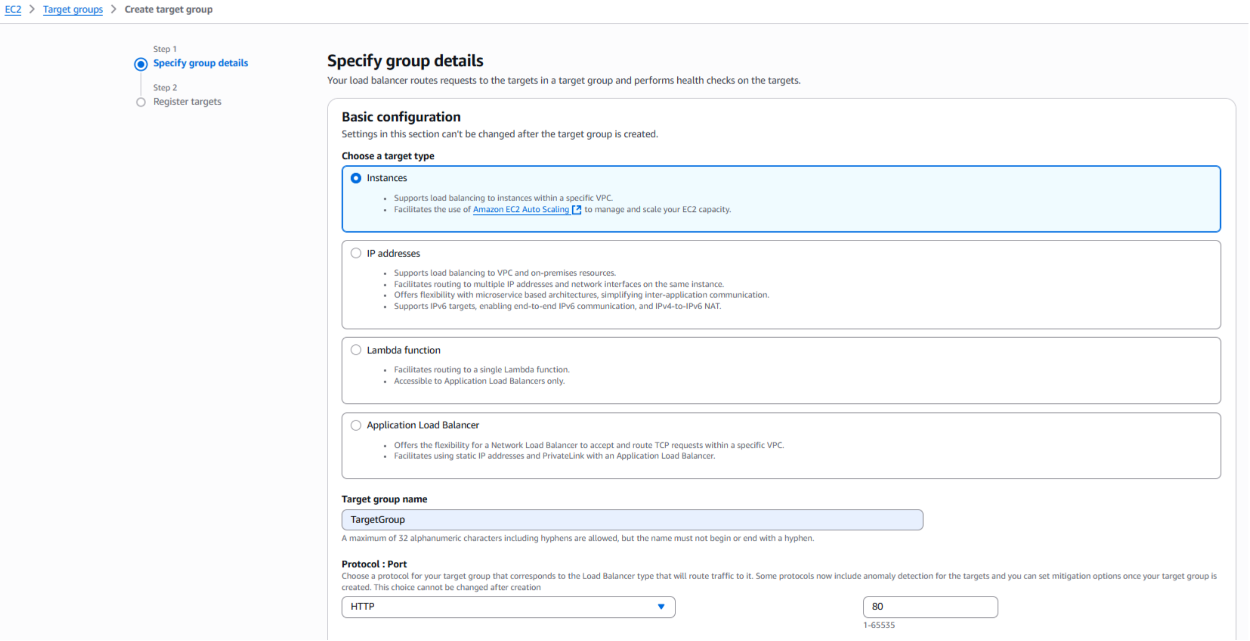
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

Add listener

9. For ***Target group name***, enter **TargetGroup**.



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.

○ Lambda function
- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

○ Application Load Balancer
- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name: TargetGroup

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

Protocol : Port

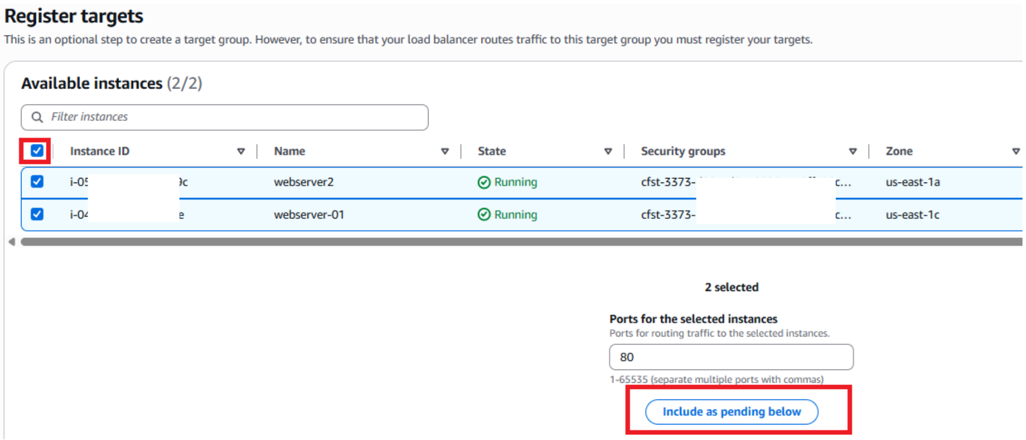
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP ▼ : 80 (1-65535)

10. Click **Next**.

11. Under ***Available instances***, select both targets that are listed.

12. Click **Include as pending below**.



Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

Available instances (2/2)

Filter instances

☑	Instance ID	Name	State	Security groups	Zone
☑	i-0[redacted]c	webserver2	✓ Running	cfst-3373-[redacted]c...	us-east-1a
☑	i-04[redacted]e	webserver-01	✓ Running	cfst-3373-[redacted]c...	us-east-1c

2 selected

Ports for the selected instances

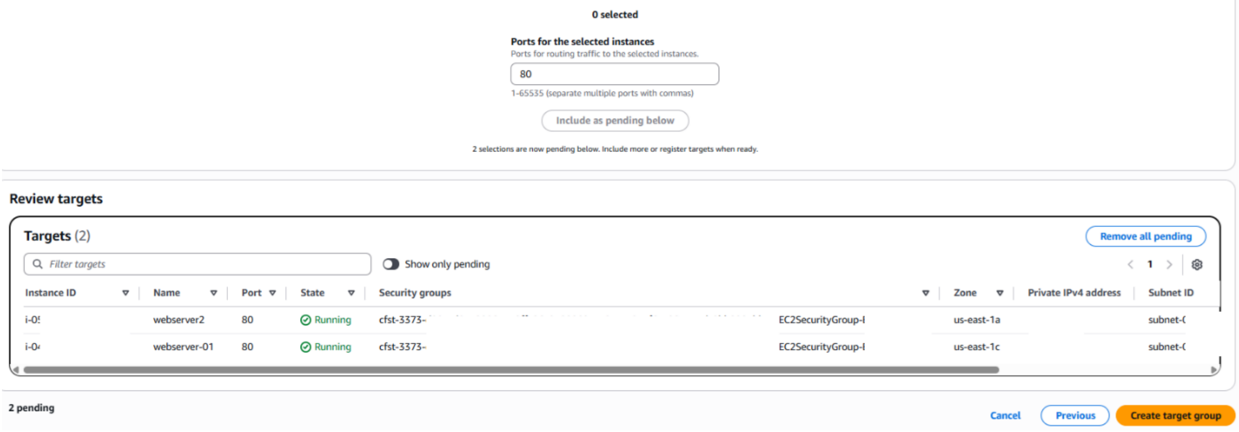
Ports for routing traffic to the selected instances.

80

1-65535 (separate multiple ports with commas)

Include as pending below

13. Click **Create target group**.



0 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

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

Review targets

Targets (2)

Remove all pending

Filter targets

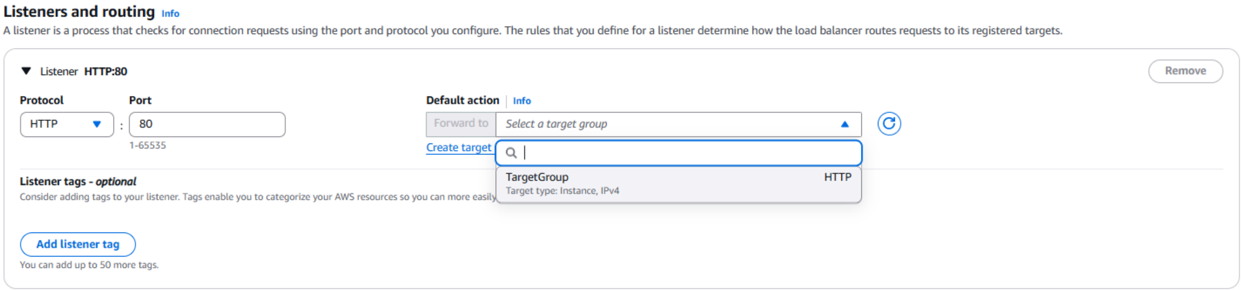
Show only pending

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0[redacted]	webserver2	80	✓ Running	cfst-3373-[redacted]EC2SecurityGroup-I	us-east-1a		subnet-(
i-04[redacted]	webserver-01	80	✓ Running	cfst-3373-[redacted]EC2SecurityGroup-I	us-east-1c		subnet-(

2 pending

Cancel Previous Create target group

14. Back in the first tab, under ***Default action***, click the **refresh** button (looks like a circular arrow), and in the dropdown, select the **TargetGroup** you just created.



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

1-65535

Default action Info

Forward to: Select a target group

Create target

TargetGroup 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

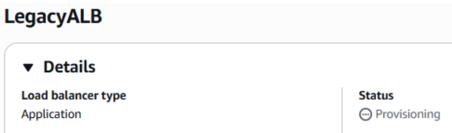
Add listener tag

You can add up to 50 more tags.

15. Click **Create load balancer**.

16. On the next screen, click **View load balancer**.

17. Wait a few minutes for the load balancer to finish provisioning and enter an active state.

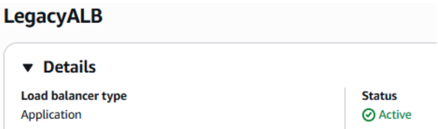


LegacyALB

▼ **Details**

Load balancer type: Application

Status: Provisioning



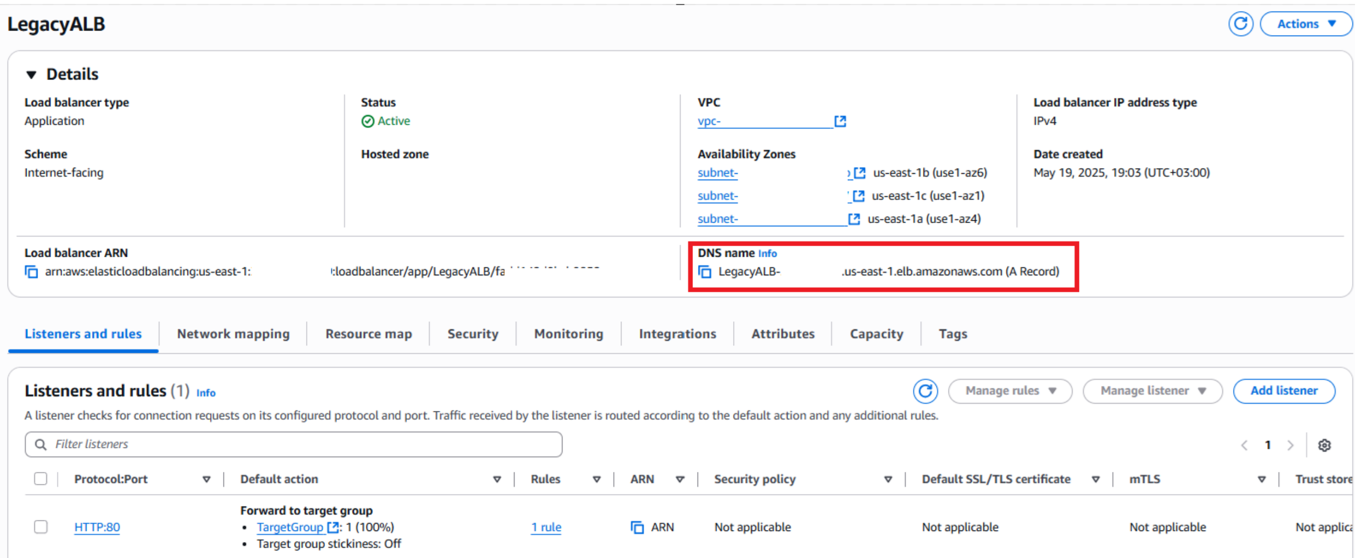
LegacyALB

▼ **Details**

Load balancer type: Application

Status: ✓ Active

18. Copy its **DNS name**, and paste it into a new browser tab. You should see the load balancer demo page again. The local IP lets you know which instance you were sent (or "load balanced") to.



LegacyALB

▼ Details

Load balancer type
Application

Scheme
Internet-facing

Status
Active

Hosted zone

VPC
vpc-[redacted]

Availability Zones
subnet-[redacted] us-east-1b (use1-az6)
subnet-[redacted] us-east-1c (use1-az1)
subnet-[redacted] us-east-1a (use1-az4)

Load balancer IP address type
IPv4

Date created
May 19, 2025, 19:03 (UTC+03:00)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:[redacted]:loadbalancer/app/LegacyALB/fa[redacted]

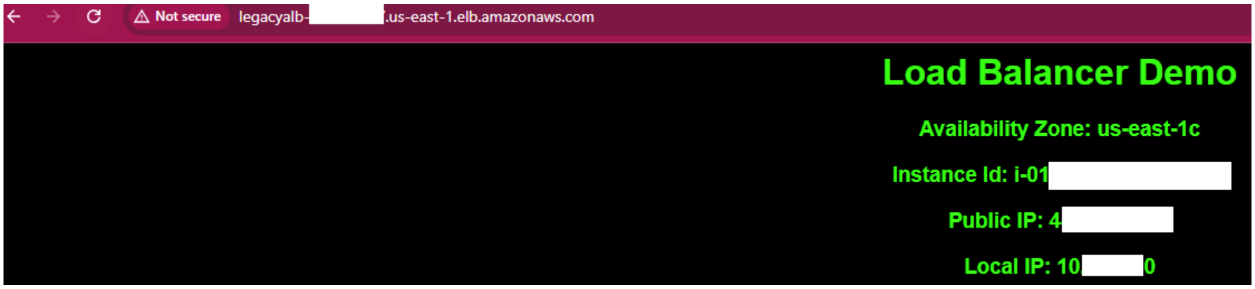
DNS name Info
LegacyALB-[redacted].us-east-1.elb.amazonaws.com (A Record)

Listeners and rules (1) Info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

□	Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store
□	HTTP:80	Forward to target group • TargetGroup: 1 (100%) • Target group stickiness: Off	1 rule	ARN	Not applicable	Not applicable	Not applicable	Not applica

19. Refresh the page a few times. You should see the other instance's local IP listed, meaning it's successfully load balancing between the two EC2 instances.



legacyalb-[redacted].us-east-1.elb.amazonaws.com

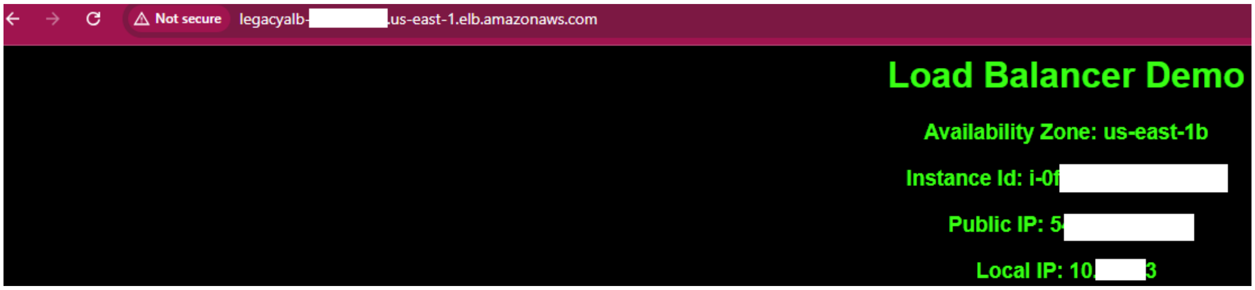
Load Balancer Demo

Availability Zone: us-east-1c

Instance Id: i-01[redacted]

Public IP: 4[redacted]

Local IP: 10[redacted]0



legacyalb-[redacted].us-east-1.elb.amazonaws.com

Load Balancer Demo

Availability Zone: us-east-1b

Instance Id: i-0f[redacted]

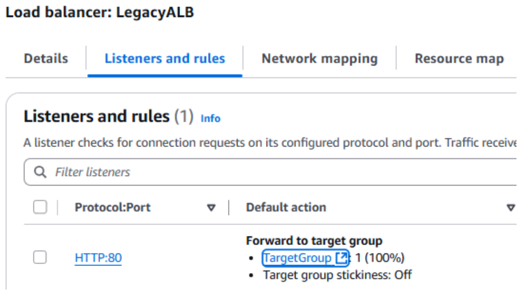
Public IP: 5[redacted]

Local IP: 10[redacted]3

20. Next, let’s enable sticky sessions, so that once we’ve connected to a server any subsequent connections will always go to the same server.

3. Enable Sticky Sessions

1. Back on the **EC2 → Load Balancers** page, select the **Listeners** tab.
2. Click the **TargetGroup** link in the ***Default action*** column, which opens the target group.



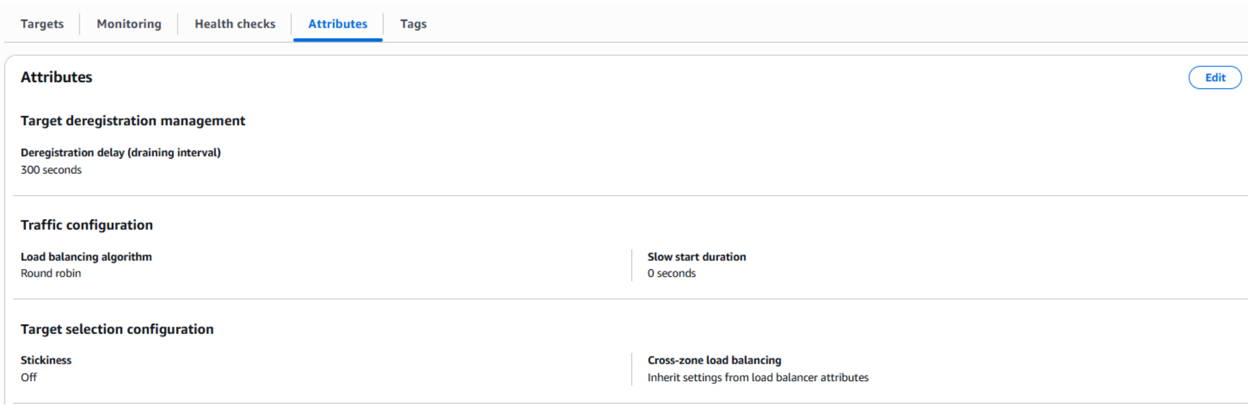
Load balancer: LegacyALB

Listeners and rules (1)

A listener checks for connection requests on its configured protocol and port. Traffic receive

□ Protocol:Port	Default action
□ HTTP:80	Forward to target group • TargetGroup 1 (100%) • Target group stickiness: Off

3. Select the **Attributes** tab. You'll notice that we have ***Stickiness*** to **Off**.
4. Click **Edit**.



Attributes

Target deregistration management

Deregistration delay (draining interval)
300 seconds

Traffic configuration

Load balancing algorithm
Round robin

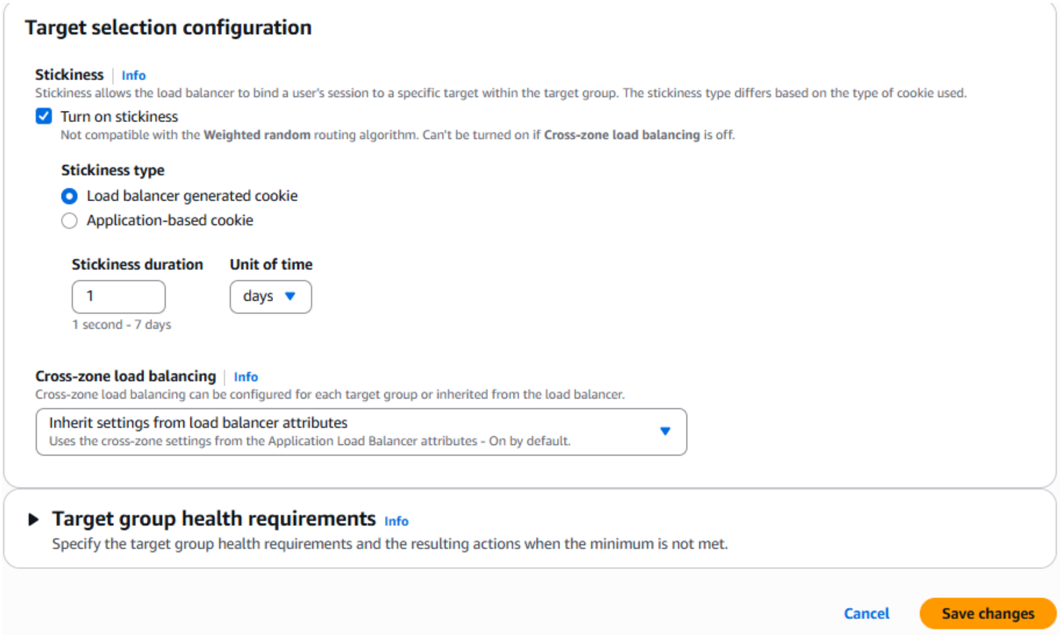
Slow start duration
0 seconds

Target selection configuration

Stickiness
Off

Cross-zone load balancing
Inherit settings from load balancer attributes

5. Check the box next to **Stickiness** to enable it (**On**).
6. Leave ***Stickiness type*** set to **Load balancer generated cookie**.
7. Leave ***Stickiness duration*** set to **1 day**.
8. Click **Save changes**.



Target selection configuration

Stickiness | Info

Stickiness allows the load balancer to bind a user's session to a specific target within the target group. The stickiness type differs based on the type of cookie used.

☑ Turn on stickiness

Not compatible with the **Weighted random** routing algorithm. Can't be turned on if **Cross-zone load balancing** is off.

Stickiness type

● Load balancer generated cookie

○ Application-based cookie

Stickiness duration	**Unit of time**
1	days
1 second - 7 days	

Cross-zone load balancing | Info

Cross-zone load balancing can be configured for each target group or inherited from the load balancer.

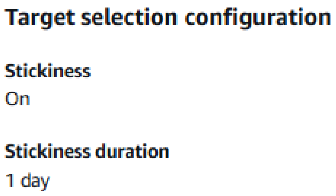
Inherit settings from load balancer attributes

Uses the cross-zone settings from the Application Load Balancer attributes - On by default.

Target group health requirements Info

Specify the target group health requirements and the resulting actions when the minimum is not met.

9. The ***Stickiness*** is **On**.

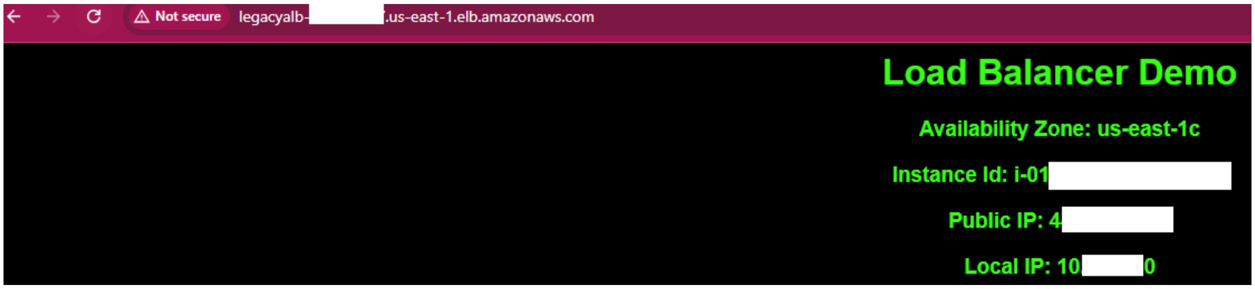


Target selection configuration

Stickiness
On

Stickiness duration
1 day

10. Refresh the tab where you navigated to the load balancer's public IP. This time, no matter how many times you refresh, it will stay on the same instance (noted by the local IP).



legacyalb-[redacted].us-east-1.elb.amazonaws.com

Load Balancer Demo

Availability Zone: us-east-1c

Instance Id: i-01[redacted]

Public IP: 4[redacted]

Local IP: 10[redacted]0