

AWS Load Balancer Setup Guide

Create instance

1. Name the instance as "web01_loa balancer"/ or anything you like
2. Select the AMI as Amazon Linux free tire
3. Instance type as t2.micro free tire
4. Create a new key pair
5. Edit security group
Security group name as
wavecafe-web

Edit Inbound traffic rules like this

Inbound Security Group Rules

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

Remove

Type	Info	Protocol	Info	Port range	Info
ssh	▼	TCP		22	
Source type	Info	Source	Info	Description - optional	Info
Anywhere	▼	<input type="text" value="0.0.0.0/0"/>	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>	

▼ Security group rule 2 (TCP, 80, 49.37.215.115/32)

Remove

Type	Info	Protocol	Info	Port range	Info
Custom TCP	▼	TCP		80	
Source type	Info	Name	Info	Description - optional	Info
My IP	▼	<input type="text" value="49.37.215.115/32"/>	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>	

Go advance details and paste this script in the comment box

```
#!/bin/bash

# Variable Declaration
#PACKAGE="httpd wget unzip"
#SVC="httpd"
URL='https://www.tooplate.com/zip-templates/2098_health.zip'
ART_NAME='2098_health'
TMPDIR="/tmp/webfiles"
```

```

yum --help &> /dev/null

if [ $? -eq 0 ]
then
    # Set Variables for CentOS
    PACKAGE="httpd wget unzip"
    SVC="httpd"

    echo "Running Setup on CentOS"
    # Installing Dependencies
    echo "#####"
    echo "Installing packages."
    echo "#####"
    sudo yum install $PACKAGE -y > /dev/null
    echo

    # Start & Enable Service
    echo "#####"
    echo "Start & Enable HTTPD Service"
    echo "#####"
    sudo systemctl start $SVC
    sudo systemctl enable $SVC
    echo

    # Creating Temp Directory
    echo "#####"
    echo "Starting Artifact Deployment"
    echo "#####"
    mkdir -p $TEMPDIR
    cd $TEMPDIR
    echo

    wget $URL > /dev/null
    unzip $ART_NAME.zip > /dev/null
    sudo cp -r $ART_NAME/* /var/www/html/
    echo

    # Bounce Service
    echo "#####"
    echo "Restarting HTTPD service"
    echo "#####"
    systemctl restart $SVC
    echo

    # Clean Up
    echo "#####"
    echo "Removing Temporary Files"

```

```

echo "#####"
rm -rf $TEMPDIR
echo

sudo systemctl status $SVC
ls /var/www/html/

else
    # Set Variables for Ubuntu
    PACKAGE="apache2 wget unzip"
    SVC="apache2"

    echo "Running Setup on CentOS"
    # Installing Dependencies
    echo "#####"
    echo "Installing packages."
    echo "#####"
    sudo apt update
    sudo apt install $PACKAGE -y > /dev/null
    echo

    # Start & Enable Service
    echo "#####"
    echo "Start & Enable HTTPD Service"
    echo "#####"
    sudo systemctl start $SVC
    sudo systemctl enable $SVC
    echo

    # Creating Temp Directory
    echo "#####"
    echo "Starting Artifact Deployment"
    echo "#####"
    mkdir -p $TEMPDIR
    cd $TEMPDIR
    echo

    wget $URL > /dev/null
    unzip $ART_NAME.zip > /dev/null
    sudo cp -r $ART_NAME/* /var/www/html/
    echo

    # Bounce Service
    echo "#####"
    echo "Restarting HTTPD service"
    echo "#####"
    systemctl restart $SVC
    echo

```

```
# Clean Up
echo "#####"
echo "Removing Temporary Files"
echo "#####"
rm -rf $TEMPDIR
echo

sudo systemctl status $SVC
ls /var/www/html/
fi
```

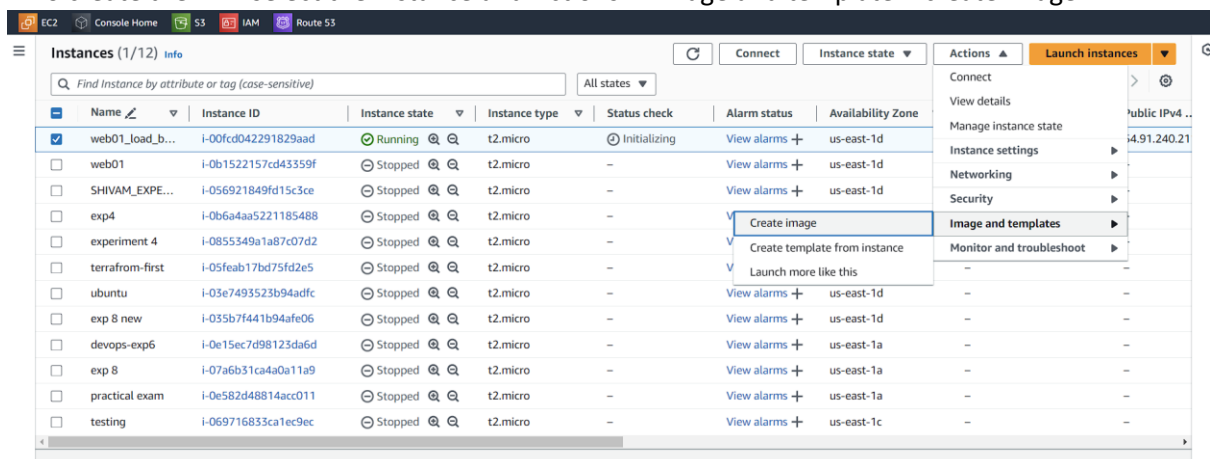
6.

- Launch the instance
and wait for some time and paste the
Public IPv4 address i.e 54.91.240.211 in browser
you will get the website

Part 2

- Now we need to create an AMI for this instance

To create the AMI select the instance and Actions >>Image and template>>create image



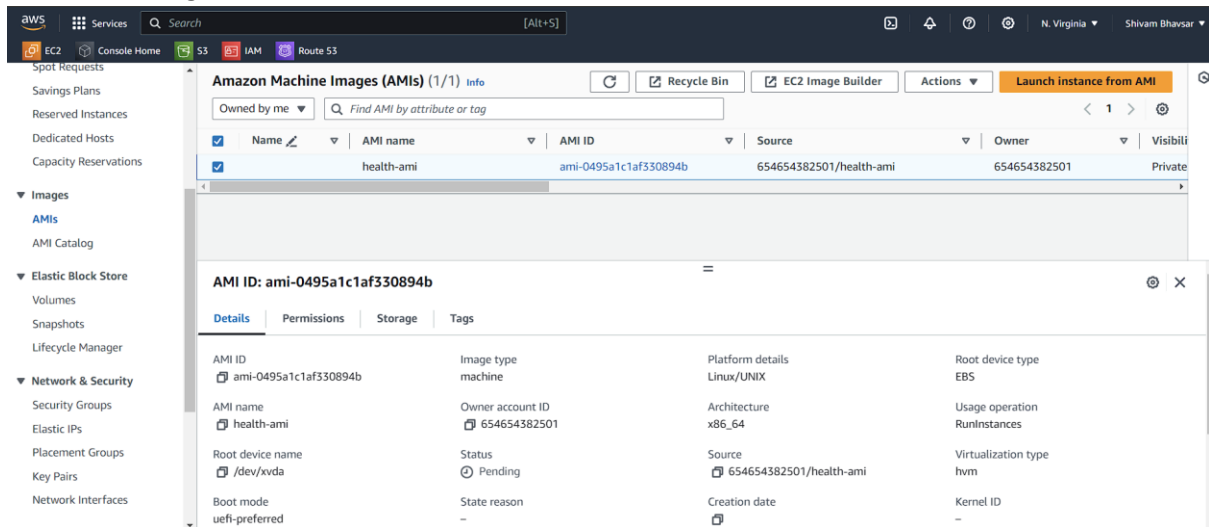
Give Image name

Change instance volume

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/...	Create new snapshot	8	EBS General Purpose	3000		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable
<button>Add volume</button>								

Click on create image

check the AMI go to ami section



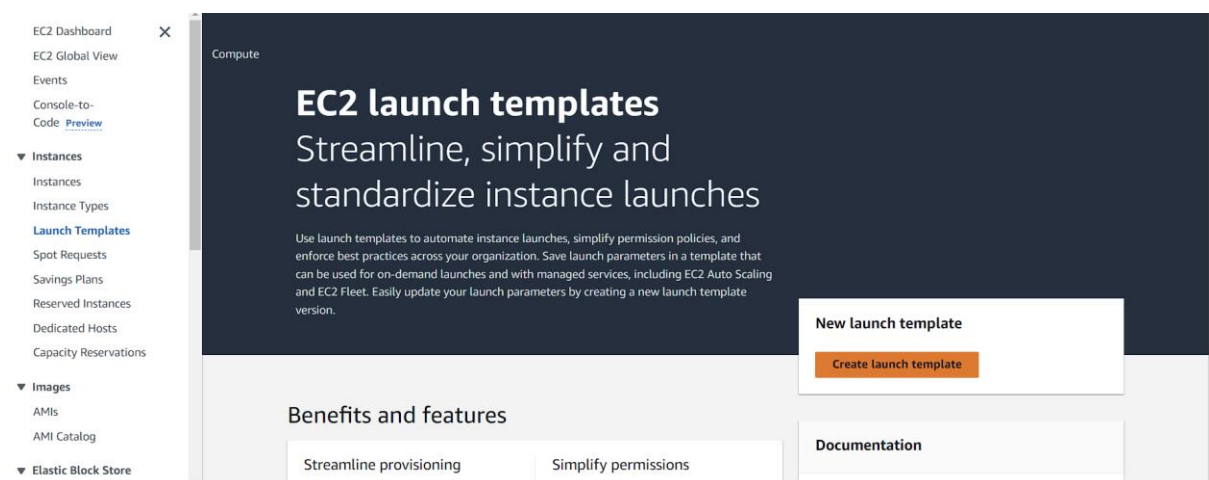
2. Click on launch the instance from AMI option on the upper right corner

Launch an instance

now instead of going through this step we will create a template so that we don't have to go through this steps

or there is an another way to create the instance by templates

3. Click on Launch Template



Click on create launch instance

4.

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

health-template

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

V1

Max 255 chars

Auto Scaling guidance [Info](#)

Select this if you intend to use this template with EC2 Auto Scaling

☐ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

Template tags

Select the AMI

▼ 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

Q

Search our full catalog including 1000s of application and OS images

Recents | **My AMIs** | Quick Start

☐ Don't include in launch template

☒ Owned by me

☐ Shared with me

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

health-ami

ami-0495a1c1af330894b

2024-07-08T14:58:39.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

▼

Select the instance type and key pair

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

Advanced

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

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ 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

wavecafe-prod-nvirl

Create new key pair

▼ Network settings [Info](#)

Subnet [Info](#)

Don't include in launch template

Create new subnet

When you specify a subnet, a network interface is automatically added to your template.

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.

Select existing security group

Create security group

Security groups [Info](#)

Select security groups

Compare security group rules

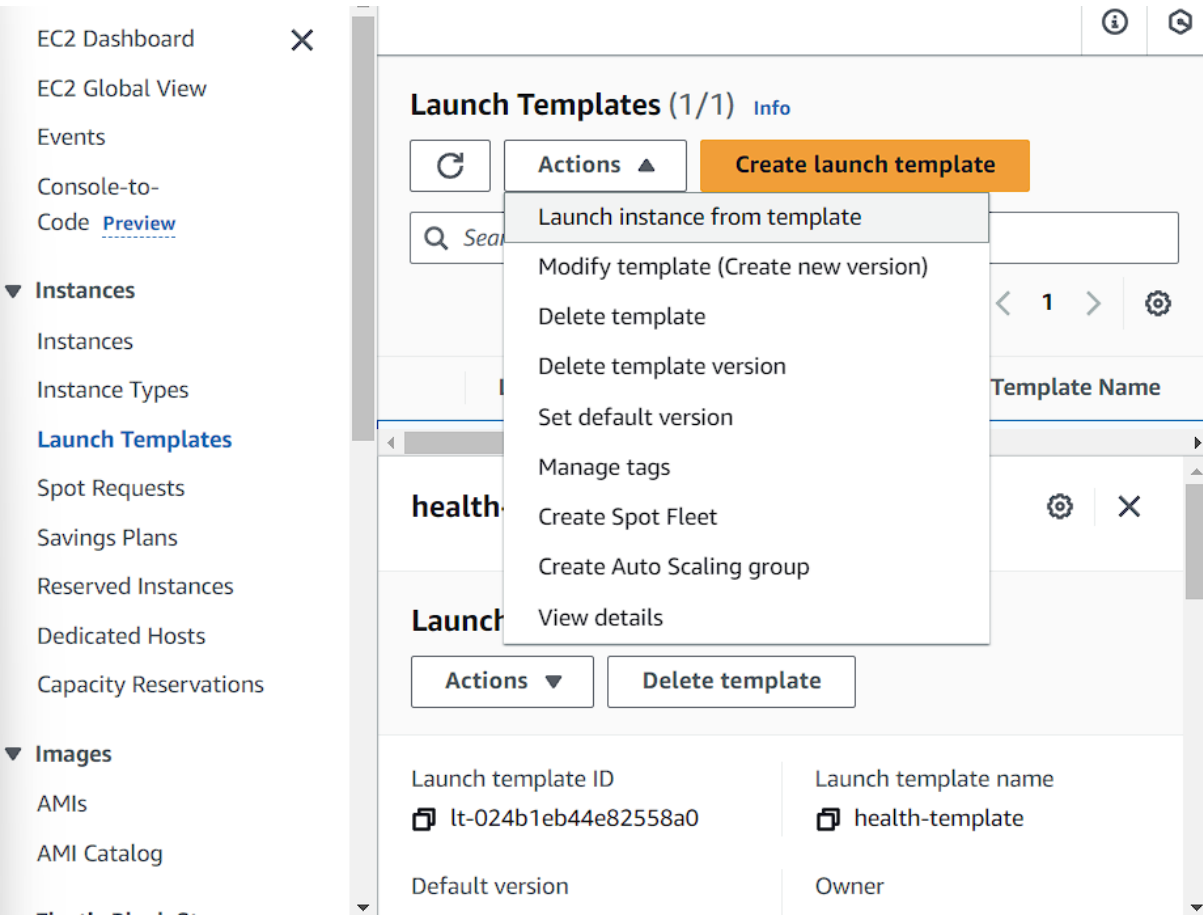
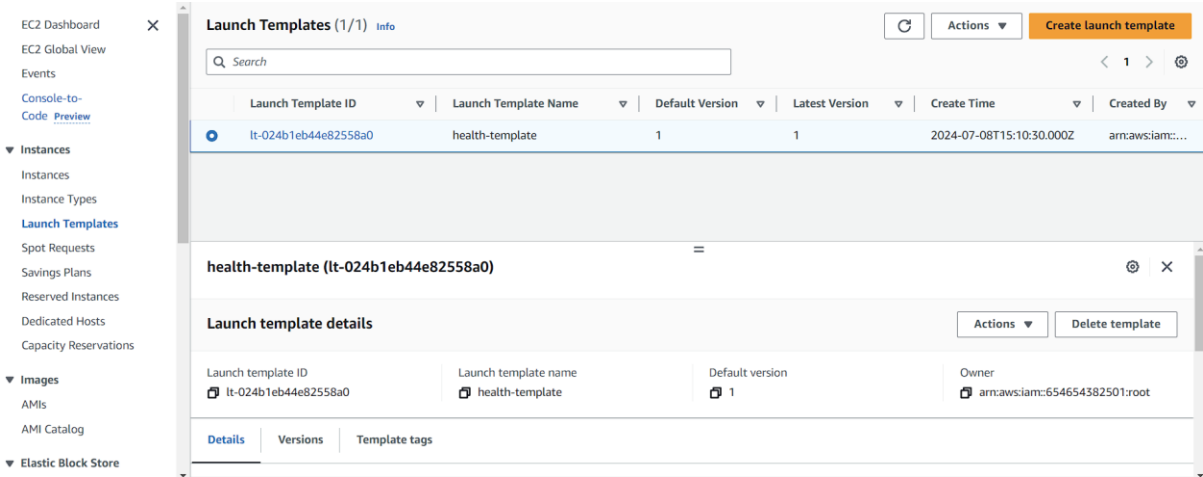
wavecafe-web sg-00b548d530667caa0 X

VPC: vpc-03e38b190720f87eb

► Advanced network configuration

Click on Create

5 Go to Launch template section and launch



Now Load Balancer

1. Go to target group , create the target group
for selection the instances

2.

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**

Target group name

health-tg

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-03e38b190720f87eb
IPv4 VPC CIDR: 172.31.0.0/16

Protocol version

☒ HTTP1

Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

☐ HTTP2

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.

/

Up to 1024 characters allowed.

▼ Advanced health check settings

Restore defaults

Health check port

The port the load balancer uses when performing health checks on targets. By default, the health check port is the same as the target group's traffic port. However, you can specify a different port as an override.

☒ Traffic port

☐ Override

group's traffic port. However, you can specify a different port as an override.

☒ Traffic port

☐ Override

Healthy threshold

The number of consecutive health checks successes required before considering an unhealthy target healthy.

2-10

Unhealthy threshold

The number of consecutive health check failures required before considering a target unhealthy.

2-10

Timeout

The amount of time, in seconds, during which no response means a failed health check.

seconds

2-120

Interval

The approximate amount of time between health checks of an individual target

seconds

5-300

Check all this and create on next

you will see the 2 instance we crated click on include as pending below

EC2 > Target groups > Create target group

Step 1
[Specify group details](#)

Step 2
Register targets

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

<input checked="" type="checkbox"/>	Instance ID	Name	State	Security groups
<input checked="" type="checkbox"/>	i-0579cbcf99273549	web02	Running	wavecafe-web
<input checked="" type="checkbox"/>	i-00fcd042291829aad	web01_load_balancer	Running	wavecafe-web

2 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.

80

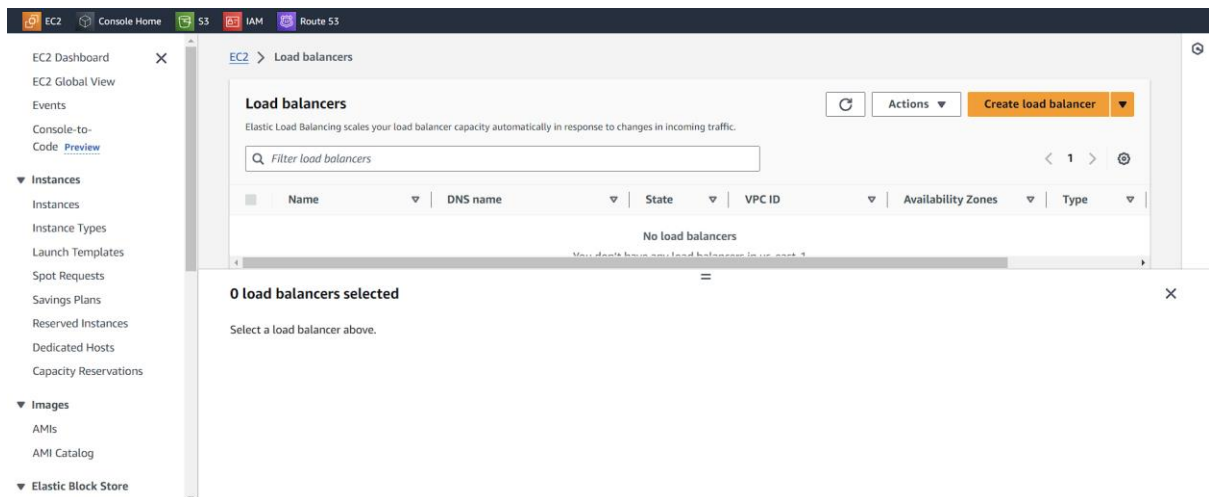
1-65535 (separate multiple ports with commas)

Include as pending below

Now click on Target group button.

Creating the load balancer

1.

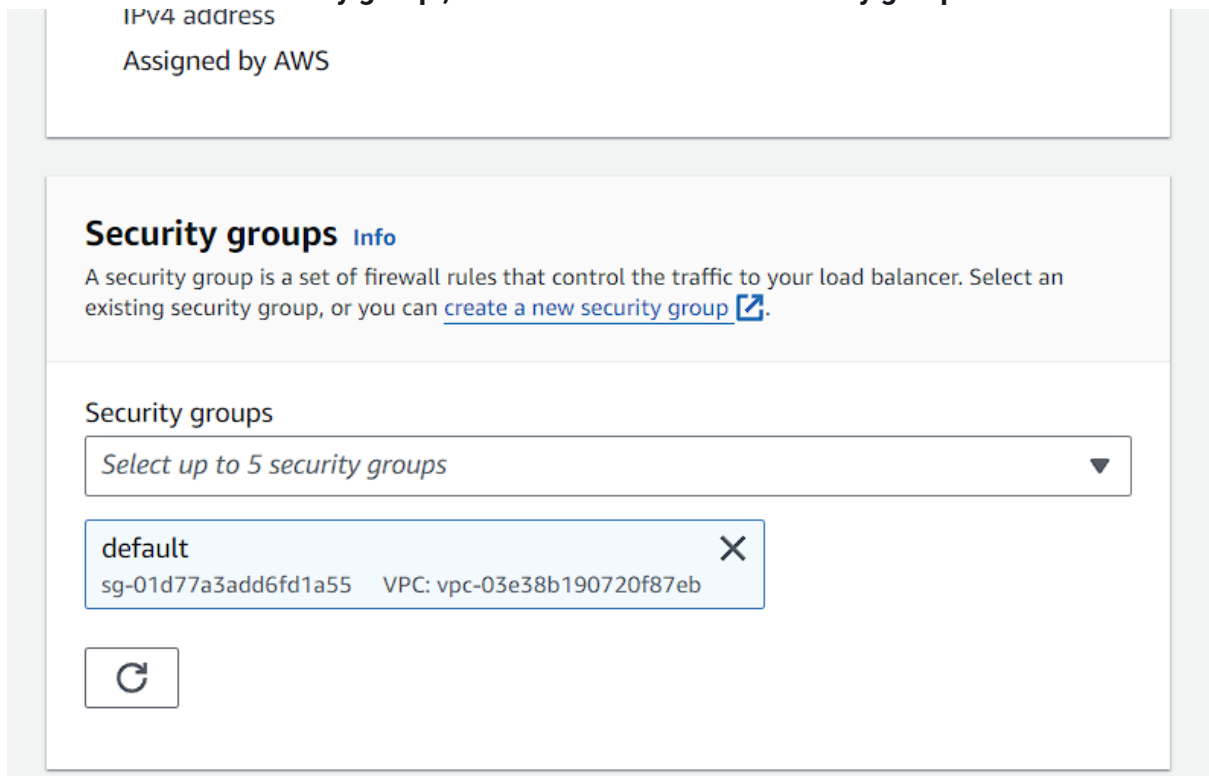


Click on create load balancer

we will choose

Application Load Balancer >> Create

2. Now create the security group , click on create the new security group



Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name [Info](#)

Name cannot be edited after creation.

Description [Info](#)

VPC [Info](#)

Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info	Description - optional	
Custom TCP ▼	TCP	80	Anywh... <input type="text" value="0.0.0.0/0"/>		Delete
Custom TCP ▼	TCP	80	Anywh... <input type="text" value="::/0"/>		Delete

[Add rule](#)

Type Info	Protocol Info	Port range Info	Destination Info	Description - optional	
All traffic ▼	All	All	Custom <input type="text" value="0.0.0.0/0"/>		Delete

[Add rule](#)

Rules with destination of 0.0.0.0/0 or ::/0 allow all IP addresses to leave the instance. We recommend setting security group rules to leave the instance from known IP addresses only. [X](#)

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#)

[Create security group](#)

3. Change the security group

Now refresh the security group and add the security group we just created

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

health-elb-sg

sg-0eaa2ce5d4a7a86ea VPC: vpc-03e38b190720f87eb

×

↻

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

Port

Default action

[Info](#)

HTTP ▼

:

80

1-65535

Forward to

health-tg

HTTP ▼

Target type: Instance, IPv4



[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.

aws

Services

Search

[Alt+S]

EC2

Console Home

S3

IAM

Route 53

N. Virginia

Shivam B

Summary

Review and confirm your configurations. [Estimate cost](#)

Basic configuration [Edit](#)

health-elb

- Internet-facing
- IPv4

Security groups [Edit](#)

- health-elb-sg
 - [sg-0eaa2ce5d4a7a86ea](#)

Network mapping [Edit](#)

VPC [vpc-03e38b190720f87eb](#)

- us-east-1a
 - [subnet-097e282be3137e2f6](#)
- us-east-1b
 - [subnet-0e0f2c41ccff56ab2](#)
- us-east-1c
 - [subnet-089c96ca888339a1e](#)
- us-east-1d
 - [subnet-0c4f87ed1a02b60a6](#)
- us-east-1e
 - [subnet-09a57b8657e01c6ce](#)
- us-east-1f
 - [subnet-02d499ed83906edd8](#)

Listeners and routing [Edit](#)

- HTTP:80 defaults to [health-tg](#)

Service integrations [Edit](#)

AWS WAF: None

AWS Global Accelerator: None

Tags [Edit](#)

None

Attributes

[subnet-09a57b8657e01c6ce](#)

- us-east-1f
- [subnet-02d499ed83906edd8](#)

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

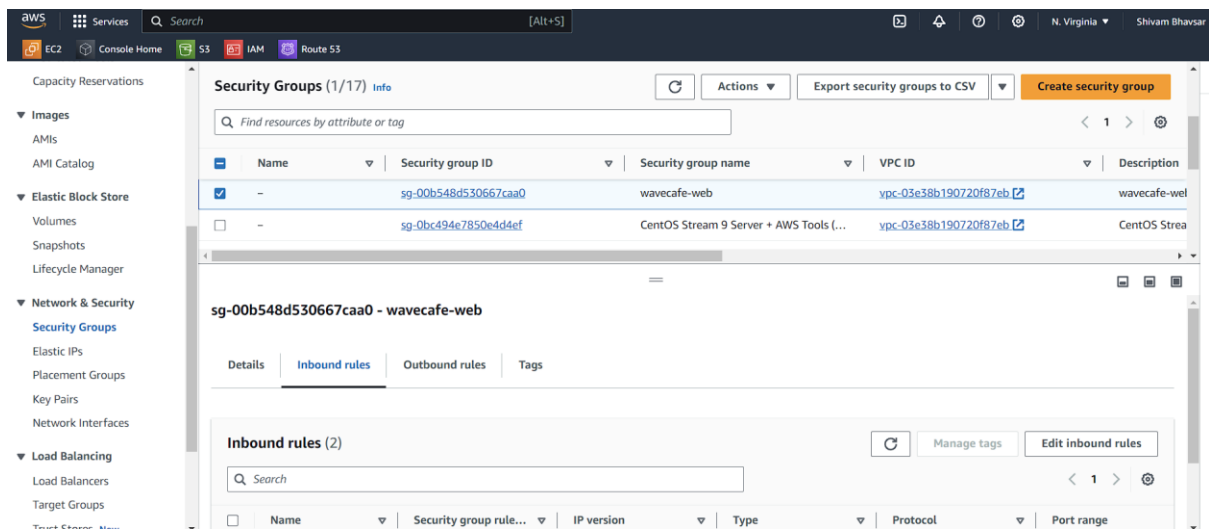
Create load balancer

Click on Create

Here it is

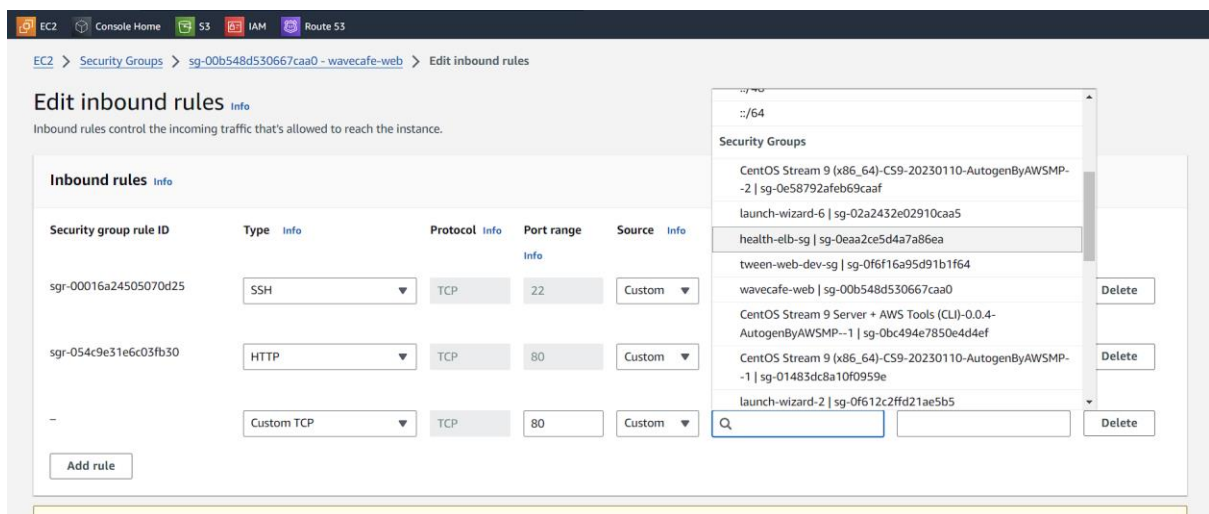
The screenshot displays the AWS Management Console interface. On the left, a sidebar contains navigation links for various AWS services, including 'EC2', 'IAM', 'Route 53', and 'Load Balancing'. The main content area is titled 'Load balancers (1/1)' and features a table with columns for Name, DNS name, State, VPC ID, Availability Zones, and Type. A single entry, 'health-elb', is listed with a status of 'Provisioning'. Below the table, the 'Details' tab for the 'health-elb' load balancer is expanded, showing a grid of key-value pairs: Load balancer type (Application), Status (Provisioning), VPC (vpc-03e38b190720f87eb), and Load balancer IP address type (IPv4).

Now go to Security Group find the security group



Select the sg
Go to inbound rule >> edit the inbound rule

GO TO ADD RULES AND ADD A RULE



4. Go to target group>>under the **Target group: health-tg**

Section>>Targets

The screenshot shows the AWS Management Console interface. On the left, the navigation menu is visible with categories like Volumes, Snapshots, Lifecycle Manager, Network & Security, Load Balancing, and Auto Scaling. The main content area displays the 'Target groups (1/1)' page. A table lists the target groups, with 'health-tg' selected. Below this, the 'Target group: health-tg' details are shown, including 'Registered targets (2)'. A table lists the registered targets, showing two instances: 'web02' and 'web01_load_b...'. Both targets are in a 'Healthy' state.

Name	ARN	Port	Protocol	Target type	Load balancer
health-tg	arn:aws:elasticloadbalanci...	80	HTTP	Instance	None associated

Instance ID	Name	Port	Zone	Health status	Health status details	Launch...
i-0579cbcf99273549	web02	80	us-east-1d	Healthy	-	July 8, 20...
i-00fcd042291829aad	web01_load_b...	80	us-east-1d	Healthy	-	July 8, 20...

You should see the healthy.

5. Go to load balancer

The screenshot shows the AWS Management Console interface. On the left, the navigation menu is visible. The main content area displays the 'Load balancers (1/1)' page. A table lists the load balancers, with 'health-elb' selected. Below this, the 'Load balancer: health-elb' details are shown, including the 'Load balancer ARN' and the 'DNS name info'.

Name	DNS name	State	VPC ID	Availability Zones	Type
health-elb	health-elb-1320996937.us...	Active	vpc-03e38b190720f8...	6 Availability Zones	application

Load balancer ARN: `arn:aws:elasticloadbalancing:us-east-1:654654382501:loadbalancer/app/health-elb/b0761536f23055cc`

DNS name info: `health-elb-1320996937.us-east-1.elb.amazonaws.com (A Record)`

Load balancers (1/1) Actions Create load balancer

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

Filter load balancers

<input checked="" type="checkbox"/>	Name	State	VPC ID	Availability Zones	Type
<input checked="" type="checkbox"/>	health-elb	Active	vpc-03e38b190720f8...	6 Availability Zones	application

Load balancer: health-elb

Copy the DNS name and paste in the browser

You will see the website is working

