

## Create Application Load Balancer

1. Navigate to EC2 Load Balancer and click on Create Load Balancer
2. In the Load Balancer type choose Application Load Balancer and click on Create option

3. Provide the load balancer name, select Internet-facing, and IPv4 address type

A:

### 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 Elastic Load Balancing works**

**Basic configuration**

Load balancer name

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

application load balancer 1

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](#)

4. In the network mapping select your VPC

5. In the subnet mapping select the availability zones and select public subnets that you have created from the dropdown

6. In the Security groups, click on create new security group and create as inbound and outbound rules as below

A:

[Alt+S]

Mumbai

gujjapranaykumar @ 6281-3291-6033

ALB securitygroup

sg-00de01fcc1661c548

application load balancer security group

vpc-0358873cfff60846f

Owner

628132916033

Inbound rules count

3 Permission entries

Outbound rules count

3 Permission entries

Inbound rules

Outbound rules

Tags

Inbound rules (3)

Manage tags

Edit inbound rules

Search

Type	Protocol	Port range	Source	Description
HTTP	TCP	80	0.0.0.0/0	-
All ICMP - IPv4	ICMP	All	0.0.0.0/0	-
HTTPS	TCP	443	0.0.0.0/0	-

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.

☒ ap-south-1a (aps1-az1)

Subnet

subnet-0434e172595c3cefc

public subnet

IPv4 address

Assigned by AWS

☐ ap-south-1b (aps1-az3)

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

ALB securitygroup

sg-00de01fcc1661c548

VPC: vpc-0358873cfff60846f

7. Once you create, select the security group and map it

8. In the Listeners and Routing, click on Create target group, select your VPC, provide target group name from the list. Other options can be default.

9. In the Advanced health check settings you can give your custom values in the traffic port. (you can leave as default if you don't wish to change it)

10. Click on 'Next' option and in the List of Registered Instances select your public instances where your nginx server runs, click 'include as pending now' option and then Create the Target Group

A:

The screenshot shows the AWS Management Console interface for 'Target groups'. At the top, there's a search bar and a 'Create target group' button. Below, a table lists the target groups. The first entry is 'albtargroup' with an ARN of 'arn:aws:elasticloadbalancing:ap-south-1:628132916033:loadbalancer/app/albtargroup', port 80, protocol HTTP, and target type Instance. Below the table, a detailed view for 'albtargroup' is shown. It includes fields for 'Instance', 'HTTP: 80', 'HTTP1', and a VPC ID 'vpc-0358873cfff60846f'. It also shows 'IP address type' as 'IPv4' and 'Load balancer' as 'None associated'. At the bottom, a summary of targets is provided: 1 Total targets, 0 Healthy, 0 Unhealthy, 1 Unused, 0 Initial, and 0 Draining.

11. Map the Target group in the Load balancer configuration

12. Now click on Create Load Balancer and it should now be created successfully

13. Navigate to Listeners tab in the Load balancer click in the Target group, select your target

A:

The screenshot shows the AWS Management Console interface for 'Load balancers'. At the top, there's a search bar and a 'Create load balancer' button. Below, a table lists the load balancers. The first entry is 'applicationloadbalancer1' with a DNS name of 'applicationloadbalancer1-247418314.ap-south-1.elb.amazonaws.com', state 'Provisioning..', VPC ID 'vpc-0358873cfff60846f', and 2 Availability Zones. Below the table, a detailed view for 'applicationloadbalancer1' is shown. It includes fields for 'south-1a (aps1-az1)', 'subnet-0df08dbebdd368656', 'ap-south-1b (aps1-az3)', 'Load balancer ARN', and 'DNS name'. The ARN is 'arn:aws:elasticloadbalancing:ap-south-1:628132916033:loadbalancer/app/applicationloadbalancer1/548dd74f64b2550c' and the DNS name is 'applicationloadbalancer1-247418314.ap-south-1.elb.amazonaws.com (A Record)'.

14. Navigate to target group and you can see the details of the targets

15. You can launch the Load Balancer using the DNS name as marked below

16. Once you access the Load balancer DNS name the web server page should be displayed on the basis of round robin schedule which is been configured in Load balancer

17. Now stop the mattermost or nginx service running in any one of the servers and check the health of the Target instance. Now you can see the unhealthy status being updated

20. Now again reload the DNS of Load balancer you should be able to access the Other healthy server

A:

The screenshot shows the AWS Management Console interface for Target groups. The top navigation bar includes the user's name 'gujjapranaykumar @ 6281-3291-6033' and the region 'Mumbai'. The main content area is titled 'Target groups (1/1)' and includes a search bar and a 'Create target group' button. Below this is a table with columns: Name, ARN, Port, Protocol, and Target type. The table contains one entry: 'albtargroutgroup' with ARN 'arn:aws:elasticloadbalanci...', Port '80', Protocol 'HTTP', and Target type 'Instance'. Below the table, there is a section titled 'Target group: albtargroutgroup' which displays a summary of target health: 2 Total targets, 2 Healthy, 0 Unhealthy, 0 Unused, 0 Initial, and 0 Draining. A sub-section titled 'Distribution of targets by Availability Zone (AZ)' is also visible, with a note to select values to see corresponding filters.

After switching off the nginx in the one of the instance

This screenshot is similar to the previous one, showing the AWS Management Console for Target groups. The summary section now shows: 2 Total targets, 1 Healthy, 1 Unhealthy, 0 Unused, 0 Initial, and 0 Draining. The 'Distribution of targets by Availability Zone (AZ)' section remains the same.

After switching on the nginx

[Alt+S]

Mumbai

gujjapranaykumar @ 6281-3291-6033

EC2 > Target groups

Target groups (1/1) Info

Filter target groups

< 1 >

<input checked="" type="checkbox"/>	Name	ARN	Port	Protocol	Target type
<input checked="" type="checkbox"/>	<a href="#">albtargetchgroup</a>	arn:aws:elasticloadbalanci...	80	HTTP	Instance

Target group: albtargetchgroup

2

Total targets

2

Healthy

0

Unhealthy

0

Unused

0

Initial

0

Draining

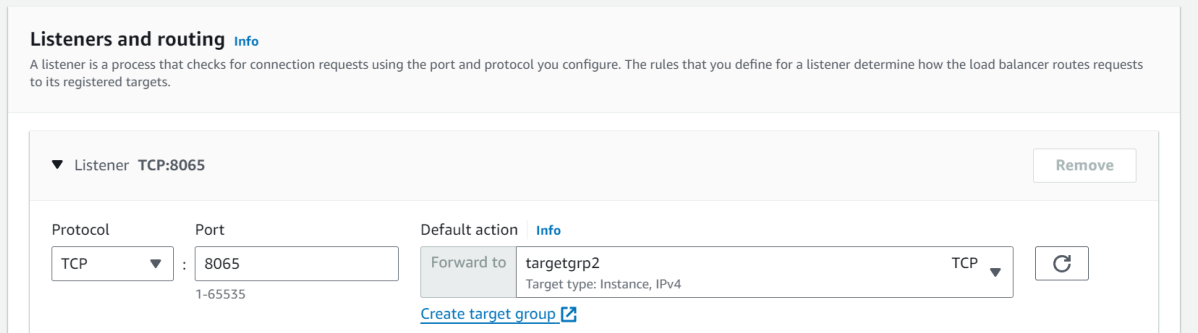
0 Anomalous

► Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Create Network Load Balancer and configure the DNS in your web server

- 1.Navigate to EC2 Load Balancer and click on Create Load Balancer
  - 2.In the Load Balancer type choose Network Load Balancer and click on Create option
  - 3.Provide the load balancer name, select Internal-facing, and IPv4 address type
  - 4.In the network mapping select your VPC
  - 5.In the subnet mapping select the availability zones and select the private subnets where you run the two mattermost instance
  - 6.In the Listeners and Routing provide the TCP port 8065 , click on Create target group and select your VPC from the list, provide the target group name other options can be default.
- A:



**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 TCP:8065 Remove

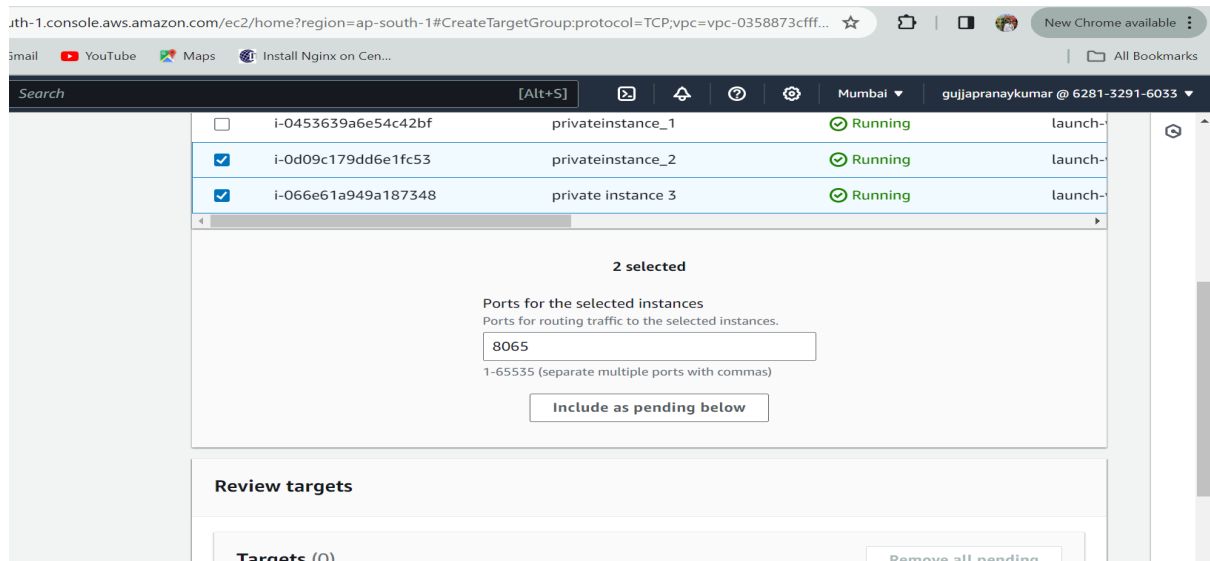
Protocol	Port	Default action	<a href="#">Info</a>
TCP ▼	: 8065 1-65535	Forward to	targetgrp2 Target type: Instance, IPv4

TCP ▼ ↻

[Create target group](#) [?](#)

- 7.In the Advanced health check settings you can give your custom values in the traffic port.(you can leave as default if you don't wish to change it)
- 8.Click on 'Next' option and in the List of Registered Instances select your private instances where you are running the mattermost
- 9.Change the port number to 8065 and 'click include as pending below' and then create the target group.

A:



10. Map the Target group in the Load balancer configuration

11. Click on Create Load Balancer and it should now be created successfully

12. Navigate to target group and you can see the details of the targets

13. Open your web server Instance , navigate to /etc/nginx/conf.d/mattermost

Provide the DNS of your Network Load Balancer

A: **pasted NLB DNS name in the both nginx configuration files**

```
upstream backend {
    server NLB1-444f5b65f22fabcf.elb.ap-south-1.amazonaws.com:8065;
    keepalive 32;
}

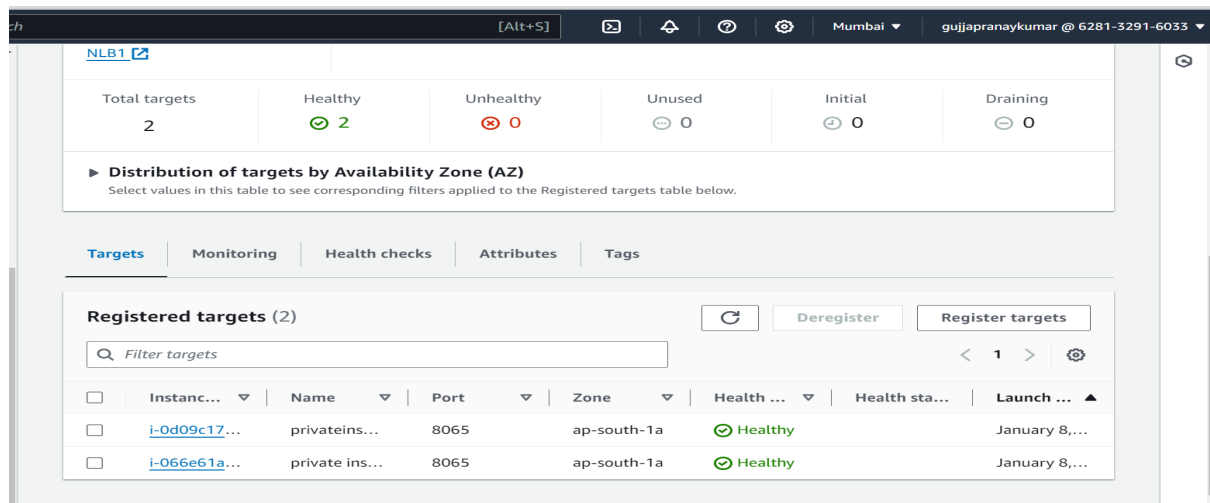
proxy_cache_path /var/cache/nginx levels=1:2 keys_zone=mattermost_cache:10m max_size=3g inactive=120m use_temp.

server {
    listen 80;
    server_name 10.0.1.106;
```

14. Now try to access your mattermost using the DNS of Application Load balancer

15. Now check the load balancing works by stopping the service / instance and also make a check on the Target health

A: before any service stop



After service stop

[Alt+S]

Mumbai

gujjapranaykumar @ 6281-3291-6033

Total targets

2

Healthy

1

Unhealthy

1

Unused

0

Initial

0

Draining

0

► Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets

Monitoring

Health checks

Attributes

Tags

Registered targets (2)

Deregister

Register targets

Filter targets

< 1 >

Instance... ▾

Name ▾

Port ▾

Zone ▾

Health s... ▾

Health stat...

Launch...

[i-0d09c179...](#)

privateinst...

8065

ap-south-1a

Unhealthy

Health che...

January 8,

[i-066e61a9...](#)

private inst...

8065

ap-south-1a

Healthy

January 8,