



**Ganpat
University**

॥ विद्यया समाजोत्कर्षः ॥

**Institute of
Computer
Technology**

Name: Tushar Panchal

En.No: 21162101014

Sub: CCE(Cloud Computing Essentials)

Branch: CBA

Batch:71

PRACTICAL 03

When designing a system, you use the principle of "design for failure and nothing will fail".

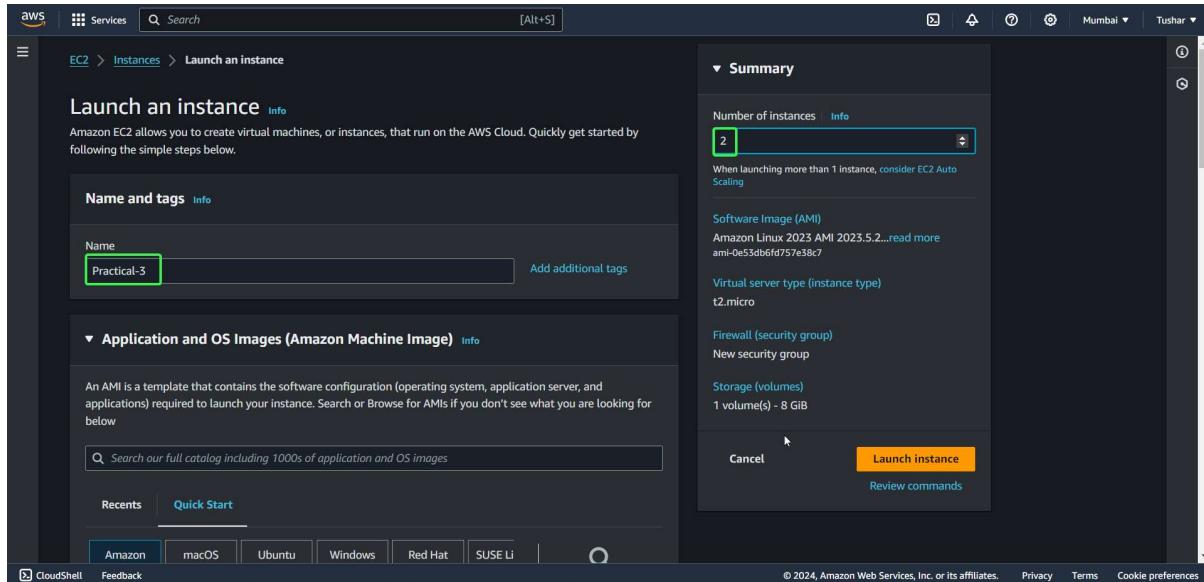
Implement the AWS Service and Resources to achieve this design principle. This lab walks you through using the Elastic Load Balancing (ELB) to load balance.

Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve fault tolerance in your applications by seamlessly providing the required amount of load-balancing capacity needed to route application traffic.

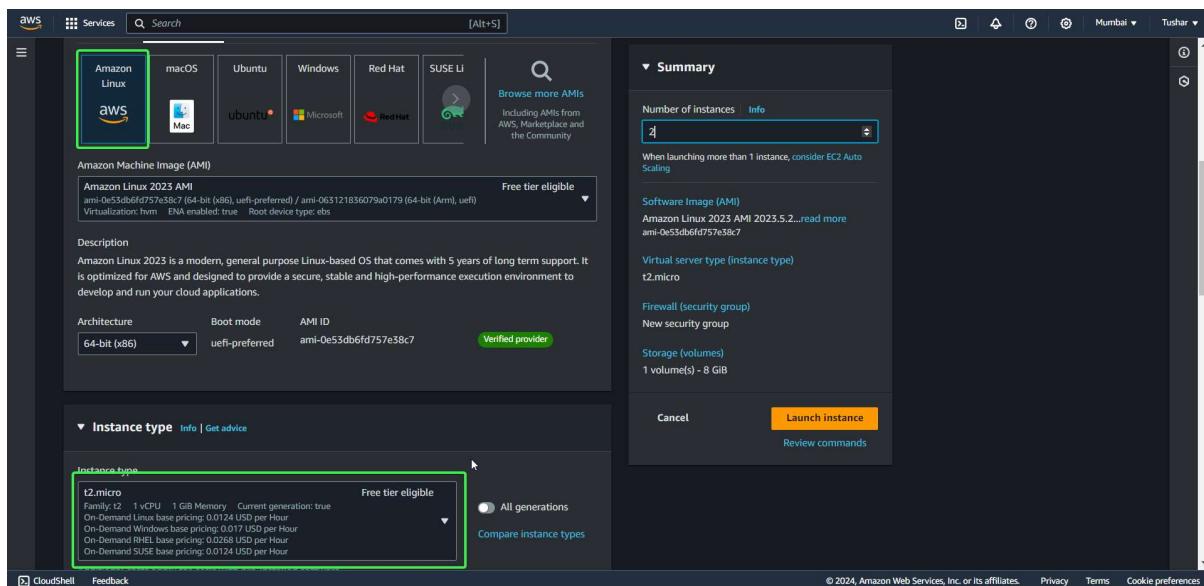
Refer scenario attached herewith & perform the following tasks:

- 1. Create Application Load Balance to balance HTTP traffic**
- 2. Create Network Load Balance to balance HTTP traffic**

» First you have to Create 2 New instances of EC2 and select the Number of instances as 2.



» Select the Amazon Linux and t2 micro instance type.



➤ Create a new Key pair.

The screenshot shows the AWS Launch Wizard interface. In the 'Key pair (login)' section, there is a dropdown menu labeled 'Select' and a button labeled 'Create new key pair'. The 'Create new key pair' button is highlighted with a green box.

The screenshot shows the 'Create key pair' dialog box. It includes fields for 'Key pair name' (set to 'Practical-3-TK'), 'Key pair type' (with 'RSA' selected), and 'Private key file format' (with '.ppk' selected). A warning message at the bottom states: 'When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance.' A 'Create key pair' button is at the bottom right.

The screenshot shows the 'Key pair (login)' section of the AWS Launch Wizard. The 'Key pair name - required' dropdown now contains the value 'Practical-3-TK', which is highlighted with a green box.

➤ After creating a key pair select the HTTP traffic from the internet and In advance add the user data. launch the instance.

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Create security group [Info](#)

Select existing security group

We'll create a new security group called 'launch-wizard-8' with the following rules:

- Allow SSH traffic from Anywhere [Info](#)
- Allow HTTPS traffic from the internet To set up an endpoint, for example when creating a web server
- Allow HTTP traffic from the internet To set up an endpoint, for example when creating a web server

⚠ Rules with source of 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.

Configure storage [Info](#) Advanced

1x 8 GiB gp3 Root volume (Not encrypted)

Add new volume

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

Metadata response top limit [Info](#)

2

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
# Use this for your user data (script from top to bottom)
# install httpd (Linux 2 version)
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>Hello World from Tushar ${hostname -f}</h1>" >
/var/www/html/index.html
```

User data has already been base64 encoded

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

➤ As You can see these two instances are created.

The screenshot shows the AWS EC2 Instances Launch log page. At the top, there is a green success message: "Successfully initiated launch of instances (i-0721271ea20fd015c, i-0444f1f77ee61ad07)". Below this, there is a "Launch log" link. Under "Next Steps", there are four cards: "Create billing and free tier usage alerts", "Connect to your instance", "Connect an RDS database", and "Create EBS snapshot policy". Each card has a "Learn more" link and a "Create" button. At the bottom, there are links for CloudShell, Feedback, and various AWS terms like Privacy, Terms, and Cookie preferences.

The screenshot shows the AWS EC2 Dashboard Instances page. On the left, there is a navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main area shows a table titled "Instances (2) Info" with the following data:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPx
Practical-3	i-0444f1f77ee61ad07	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b	ec2-3-111-
Practical-3	i-0721271ea20fd015c	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b	ec2-13-23-

Below the table, there is a "Select an instance" dropdown menu.

» 1. Create Application Load Balance to balance HTTP traffic.

The screenshot shows the AWS Management Console with the EC2 service selected. Under 'Load Balancing', 'Load Balancers' is selected. The main content area is titled 'Compare and select load balancer type' and provides a brief overview of each load balancer type: Application Load Balancer, Network Load Balancer, and Gateway Load Balancer. The Application Load Balancer is highlighted with a green border and a callout box. The Network Load Balancer and Gateway Load Balancer are also shown with their respective architectures and target types.

You have to select the ALB(Application Load Balancer).

The screenshot shows the AWS Management Console with the EC2 service selected. Under 'Load Balancing', 'Load Balancers' is selected. The main content area displays a table titled 'Load balancers' with one row showing 'No load balances'. Below the table, a message says 'You don't have any load balancers in ap-south-1'. At the top right of the table, there is a prominent orange button labeled 'Create load balancer' with a dropdown arrow. The left sidebar lists various AWS services and regions.

Now Create the load Balancer.

Gave the name of the Load balancer

The screenshot shows the 'Create Application Load Balancer' wizard. In the 'Basic configuration' step, the 'Load balancer name' field is set to 'Practical-3'. The 'Scheme' dropdown is set to 'Internet-facing'. The 'Load balancer IP address type' dropdown is set to 'Public IPv4'.

Select the two-zones South 1a and 1b

The screenshot shows the 'Mappings' section of the VPC configuration. Under 'Availability Zones', 'ap-south-1a (aps1-az1)' and 'ap-south-1b (aps1-az3)' are selected and highlighted with a green border. Other options like 'ap-south-1c (aps1-az2)' are shown but not selected.

Select the security group the same as the instance has.

The screenshot shows the AWS EC2 Instances page. There are two instances listed: 'Practical-3' (running, t2.micro, in ap-south-1b) and another 'Practical-3' (running, t2.micro, in ap-south-1b). The second 'Practical-3' instance is selected. The 'Security' tab is selected, showing the security group 'sg-04b8c1faf7ecc0c5e (launch-wizard-8)' highlighted with a green box.

The screenshot shows the AWS Load Balancer configuration page. In the 'Security groups' section, the security group 'launch-wizard-8' is selected and highlighted with a green box.

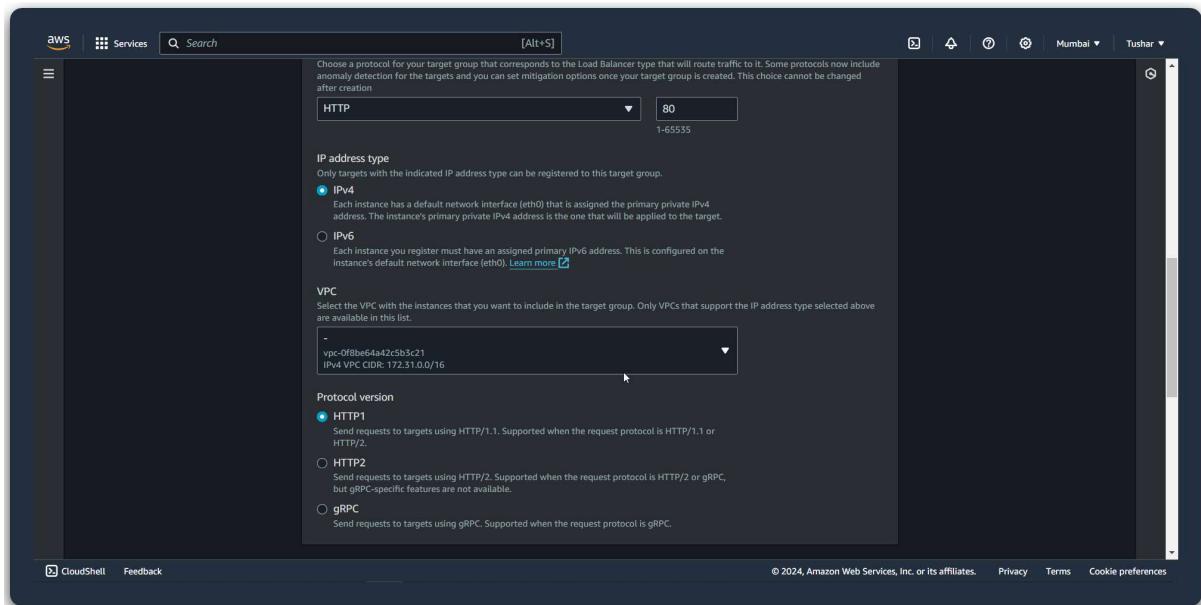
After that, we have to make a new Target Group

The screenshot shows the AWS EC2 Target Groups page. On the left sidebar, under the Load Balancing section, 'Target Groups' is selected and highlighted with a green box. At the top right, there is a yellow 'Create target group' button, which is also highlighted with a green box. The main area displays a message: 'No target groups' and 'You don't have any target groups in ap-south-1'. Below this, there is a 'Create target group' button.

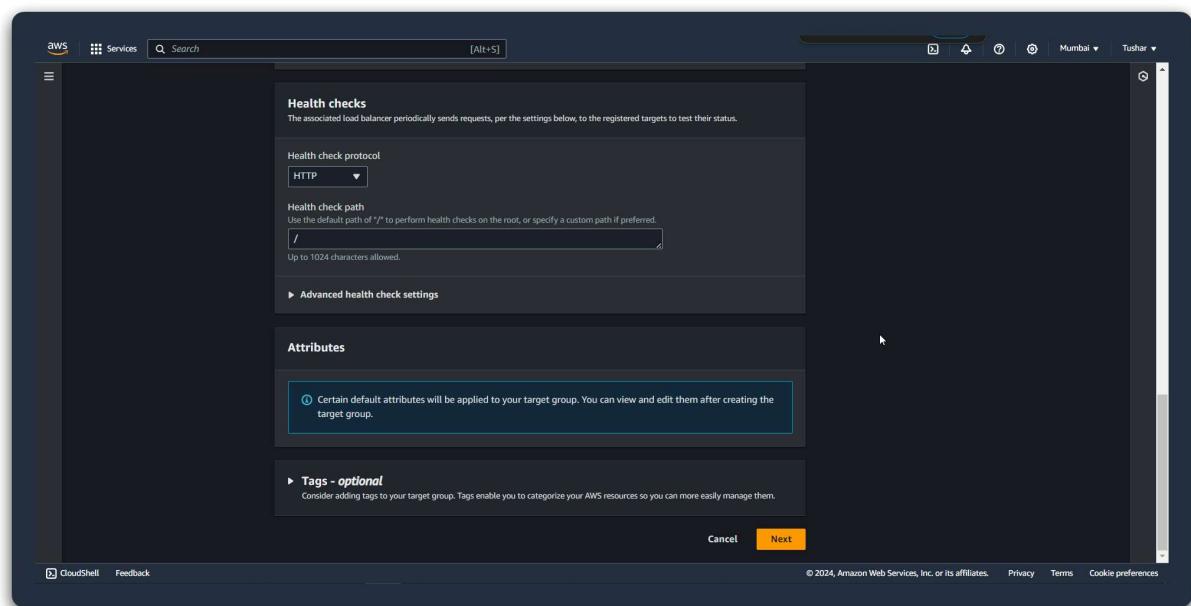
Give the target Group name.

The screenshot shows the 'Create target group' configuration page. It includes sections for selecting a target type (IP addresses, Lambda function, Application Load Balancer), a 'Target group name' field containing 'PRAC-3' (which is highlighted with a green box), and fields for 'Protocol : Port' (set to HTTP:80) and 'IP address type' (set to 1-65535). The 'Protocol : Port' section also contains a note about protocol anomalies and mitigation options.

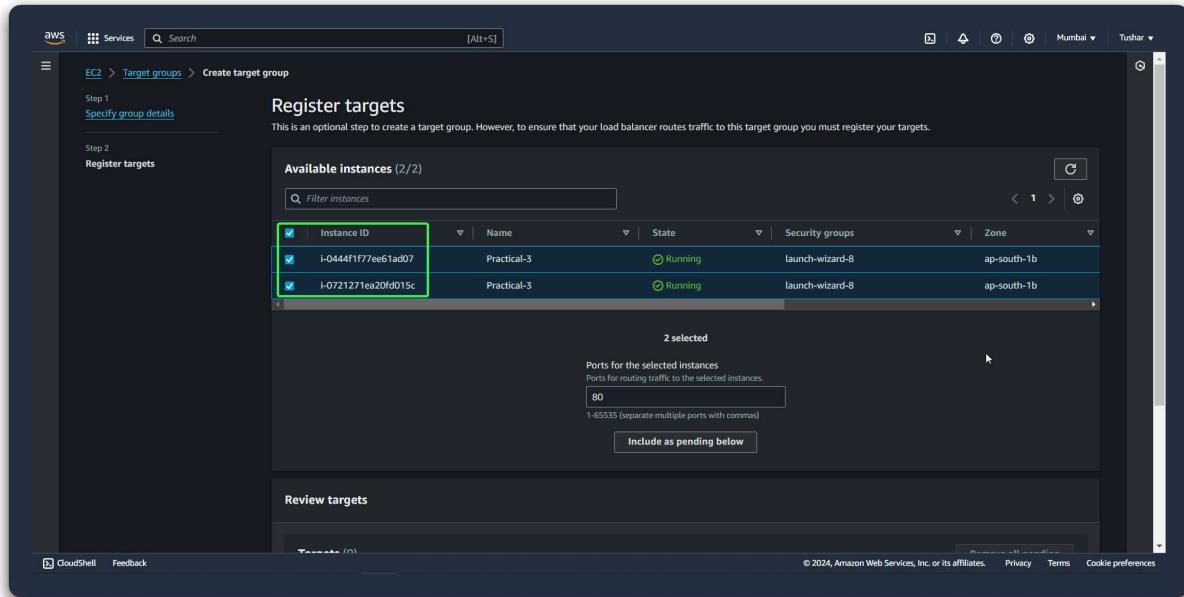
Do not change anything



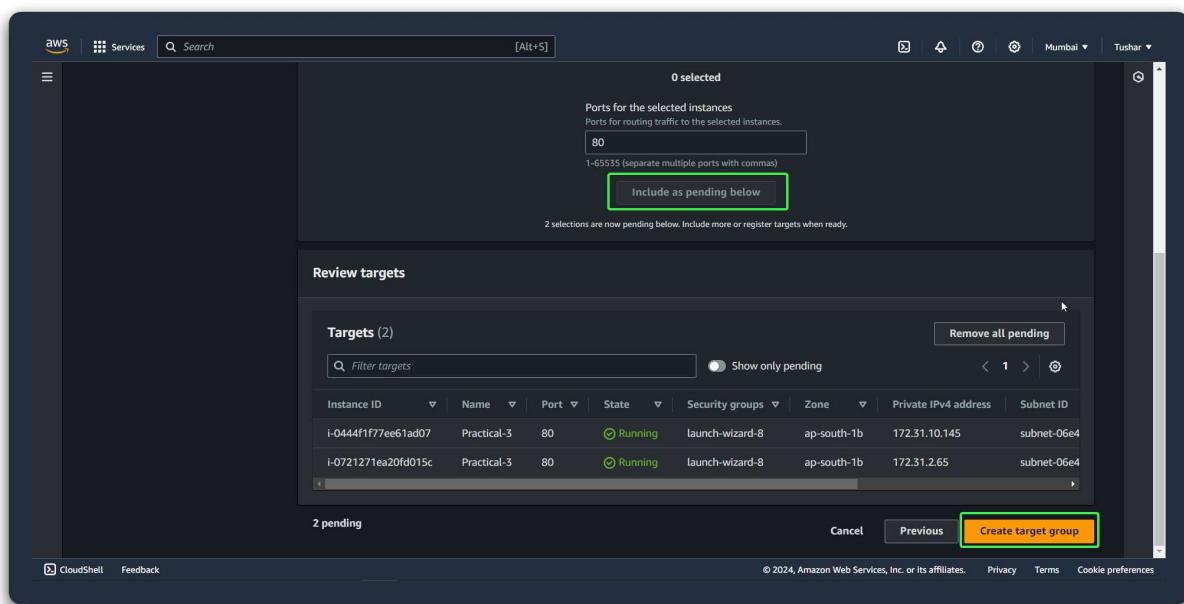
skip this part



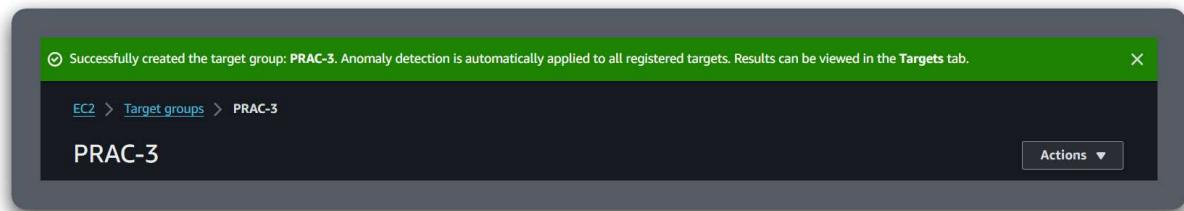
After that, You have to register the 2 instances.



And after selecting these instances create the target group.



As You can see this is the target group created



Now after creating target group, you have to select the target group you created in the load balancer creation part

The screenshot shows the AWS CloudFront 'Listeners and routing' configuration page. On the left, the navigation pane includes 'Images', 'AMIs', 'AMI Catalog', 'Elastic Block Store', 'Volumes', 'Snapshots', 'Lifecycle Manager', 'Network & Security', 'Security Groups', 'Elastic IPs', 'Placement Groups', 'Key Pairs', 'Network Interfaces', 'Load Balancing', 'Load Balancers', 'Target Groups', 'Trust Stores', and 'Auto Scaling'. The 'Target Groups' link is underlined, indicating it's the active section. In the main content area, a 'Listener HTTP:80' is selected. The 'Protocol' is set to 'HTTP' and 'Port' is '80'. The 'Default action' dropdown is open, showing 'Forward to' and a list of target groups. The 'PRAC-3' target group is highlighted with a green border. Below the dropdown, there are sections for 'Listener tags - optional' and 'Load balancer tags - optional'. At the bottom, there are buttons for 'Add listener tag' and 'Add listener'.

This screenshot shows the same AWS CloudFront configuration page as the previous one, but with a slight difference. The 'Forward to' dropdown menu is now closed, and the 'PRAC-3' target group is listed as a selected item in the dropdown menu. The rest of the interface remains the same, including the navigation pane, the 'Listener HTTP:80' configuration, and the optional tags sections.

Practical-3

Outputs

Name	Type	Description
Practical-3 URL	String	The URL of the deployed application.

Outputs - expanded

ARN	Description
arn:aws:cloudwatchlogs:us-east-1:123456789012:log-group:/aws/lambda/Practical-3	CloudWatch Metrics Log group for the Lambda function.

Review the summary and After that Create load balancer

Review the load balancer configurations and make changes if needed. After you finish reviewing the configurations, choose **Create load balancer**.

Summary

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

Basic configuration	Security groups	Network mapping	Listeners and routing
Practical-3 <ul style="list-style-type: none"> Internet-facing IPv4 	launch-wizard-8 sg-04b8c1faf1ecc0c5e	VPC vpc-0f8be64a42c5b3c21 <ul style="list-style-type: none"> ap-south-1a subnet-039d1ff0631a6e2b ap-south-1b subnet-06e4e4c81ab6594c9 	HTTP:80 defaults to PRAC-3

Service integrations

AWS WAF: None
AWS Global Accelerator: 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

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.

Create load balancer

The screenshot shows the AWS Cloud9 IDE interface. On the left, a sidebar lists various AWS services: Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, Target Groups, Trust Stores, Auto Scaling, and Auto Scaling Groups. The 'Load Balancers' section is expanded, showing 'Load balancers' and 'Target Groups'. The main content area displays the details of a load balancer named 'Practical-3'. The 'Details' tab is selected, showing the following information:

- Load balancer type:** Application
- Status:** Provisioning
- VPC:** vpc-0f8be64a42c5b3c21
- Load balancer IP address type:** IPv4
- Scheme:** Internet-facing
- Hosted zone:** ZP97RAFLXTNZK
- Availability Zones:** subnet-06e4e4c81ab6594c9 ap-south-1b (aps1-az2), subnet-059df1ff0631a6e2b ap-south-1a (aps1-az1)
- Load balancer ARN:** arn:aws:elasticloadbalancing:ap-south-1:009160043403:loadbalancer/app/Practical-3/2e9fdccfc9ab77c
- DNS name info:** Practical-3-757466816.ap-south-1.elb.amazonaws.com (A Record)

Below the details, there are tabs for 'Listeners and rules', 'Network mapping', 'Resource map - new', 'Security', 'Monitoring', 'Integrations', 'Attributes', and 'Tags'. At the bottom of the page, there are links for 'CloudShell', 'Feedback', and 'Cookie preferences'.

As you can see in above our load balancer was created successfully

Wait for State Provisioning after active go to your DNS Link

Here you can see the IP address of the First Instance.

The screenshot shows a web browser window with the URL 'practical-3-757466816.ap-south-1.elb.amazonaws.com'. The page content is 'Hello World from Tushar ip-172-31-2-65.ap-south-1.compute.internal'.

And here Is the other Ip of Another instance after reloading the page

The screenshot shows a web browser window with the URL 'practical-3-757466816.ap-south-1.elb.amazonaws.com'. The page content is 'Hello World from Tushar ip-172-31-10-145.ap-south-1.compute.internal'.

» 2. Create Network Load Balance to balance HTTP traffic.

The screenshot shows the AWS Management Console with the EC2 service selected. In the left sidebar, under 'Load Balancing', 'Load Balancers' is highlighted with a green box. The main content area displays three load balancer types: Application Load Balancer (ALB), Network Load Balancer (NLB), and Gateway Load Balancer (GWLB). The NLB section is also highlighted with a green box. Below each section is a brief description and a diagram illustrating its architecture.

Here you have to select the second one which is the Network load Balancer.

This is another screenshot of the AWS EC2 Load Balancers comparison page. It shows the same three load balancer types: ALB, NLB, and GWLB. The NLB section is highlighted with a green box, indicating it is the recommended choice for this task.

Give it a name and the same Configuration as the first we have done

Create Network Load Balancer Info

The Network Load Balancer distributes incoming TCP and UDP traffic across multiple targets such as Amazon EC2 instances, microservices, and containers. When the load balancer receives a connection request, it selects a target based on the protocol and port that are specified in the listener configuration, and the routing rule specified as the default action.

How Network 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.
PRAC-3-N/W
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme
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.

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.
 IPv4

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

Select Two zones as mine.

Mappings
Select one or more Availability Zones and corresponding subnets. Enabling multiple Availability Zones increases the fault tolerance of your applications. The load balancer routes traffic to targets in the selected Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones
 ap-south-1a (aps1-az1)
 Subnet: subnet-039df1ff0631a6e2b
 IPv4 subnet CIDR: 172.31.3.0/20

IPv4 address
The front-end IPv4 address of the load balancer in the selected Availability Zone.
 Assigned by AWS Use an Elastic IP address

ap-south-1b (aps1-az3)
 Subnet: subnet-06e4e4c81ab6594c9
 IPv4 subnet CIDR: 172.31.0.0/20

IPv4 address
The front-end IPv4 address of the load balancer in the selected Availability Zone.
 Assigned by AWS Use an Elastic IP address

ap-south-1c (aps1-az2)

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](#).

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

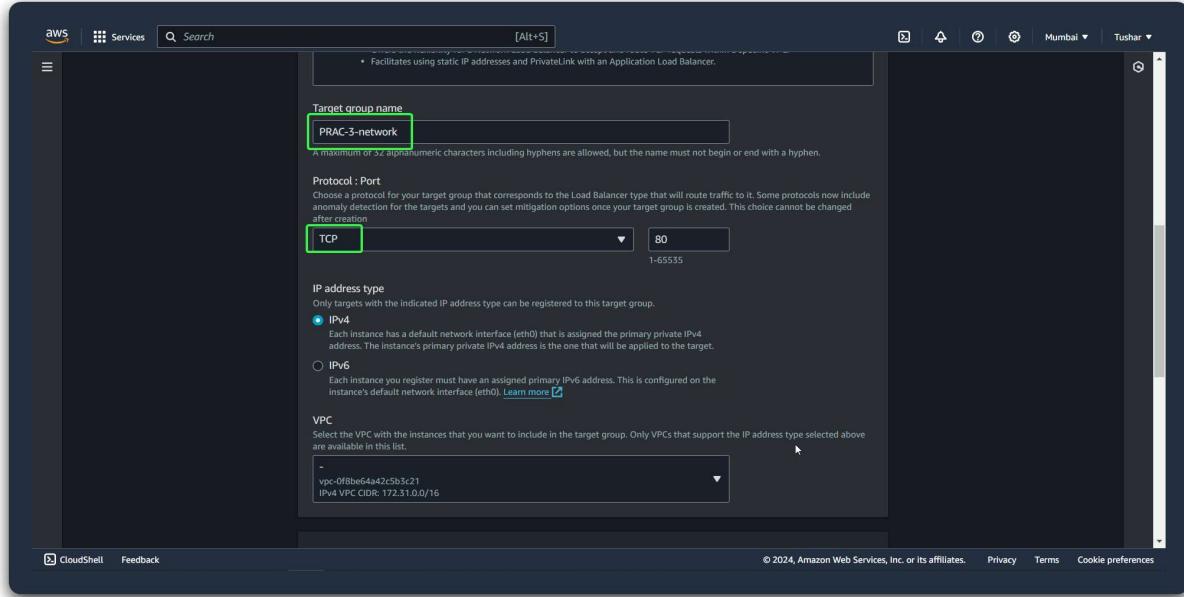
Select SG as instance as have And after that, you have to Create a Target Group

The screenshot shows the AWS Lambda console interface. A new function named "Launch-wizard-8" is being created. The "Code" tab is active, displaying the Lambda@Edge code. The "Configuration" tab is also visible.

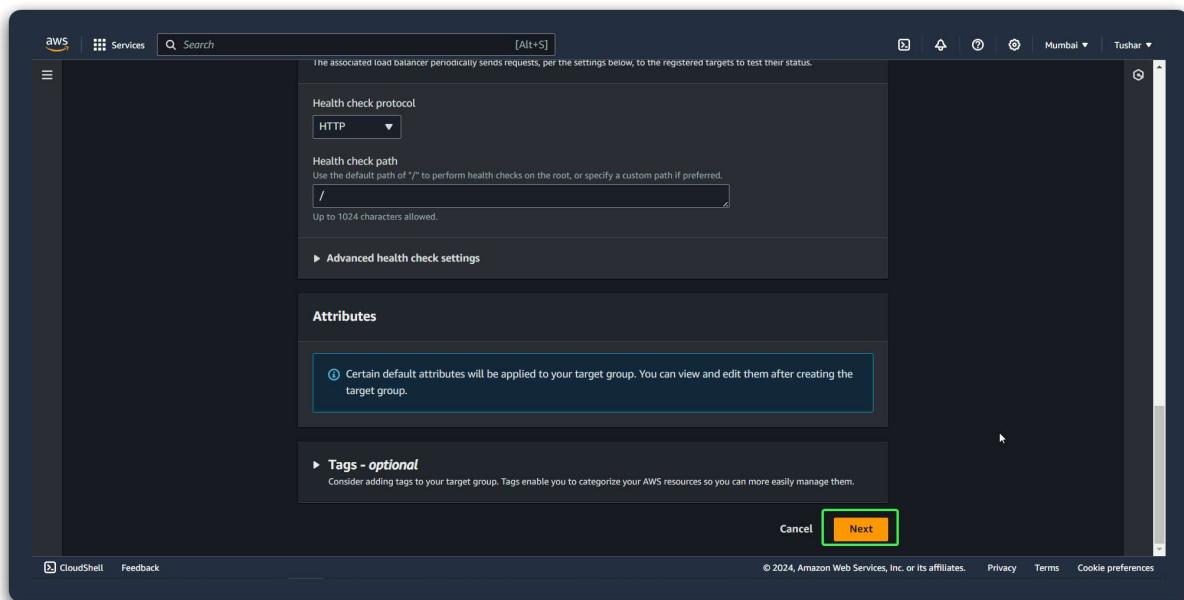
Select the type of target

The screenshot shows the AWS EC2 Target Groups console. A new target group is being created, specifically for an Application Load Balancer. The "Instances" option is selected under "Choose a target type". Other options like "IP addresses" and "Lambda function" are also shown.

Give the name of the Target Group and select TCP in the Protocol



Same as previous one and head to next



Select the Two instances and create the Target Group.

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)

Instance ID	Name	State	Security groups
i-0444f1f77ee61ad07	Practical-3	Running	launch-wizard-8
i-0721271ea20fd015c	Practical-3	Running	launch-wizard-8

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

Create the target Group.

Review targets

Targets (0)

No instances added yet
Specify instances above, or leave the group empty if you prefer to add targets later.

Create target group

As you see the new target group is created.

PRAC-3-network

Actions ▾

Now select the Target group you have created.

The screenshot shows the 'Listeners and routing' section of the AWS CloudFront configuration. A specific listener for port TCP:80 is selected. The 'Forward to' field is highlighted with a green box and contains the target group 'PRAC-3-network'. Other fields visible include 'Protocol' (TCP), 'Port' (80), and 'Default action' (Info).

After selecting head Create the load balancer and create it.

The screenshot shows the 'Create load balancer' wizard. The 'Basic configuration' tab is active, showing a VPC (vpc-0f8be64a42c5b3c21) and security groups (launch-wizard-8). The 'Listeners and routing' tab is also visible. At the bottom right, the 'Create load balancer' button is highlighted with a green box.

Here you can see the Load Balancer has Created Successfully.

The screenshot shows the AWS CloudFront dashboard. A green success message at the top states 'Successfully created load balancer'. Below it, the navigation path is EC2 > Load balancers > PRAC-3-Network. The main title 'PRAC-3-Network' is displayed, and there is a 'Actions' dropdown menu.

Here you can see the IP address of the First instance.



As you see the IP of the Second Instance is Different after
Reloading the page

