



Classic Load Balancer A Mini Project

📌 Introduction

This project demonstrates how to deploy an Classic Load Balancer(CLB) in linux. The Classic Load Balancer it act like Round-Robin traffic distribution method. The classic load balancer is used in monolithic architecture website. The main objective of this project was to learn how to deploy classic load balancer. Finally, The CLB name of the classic load balancer is copied and tested to verify traffic disributtion.



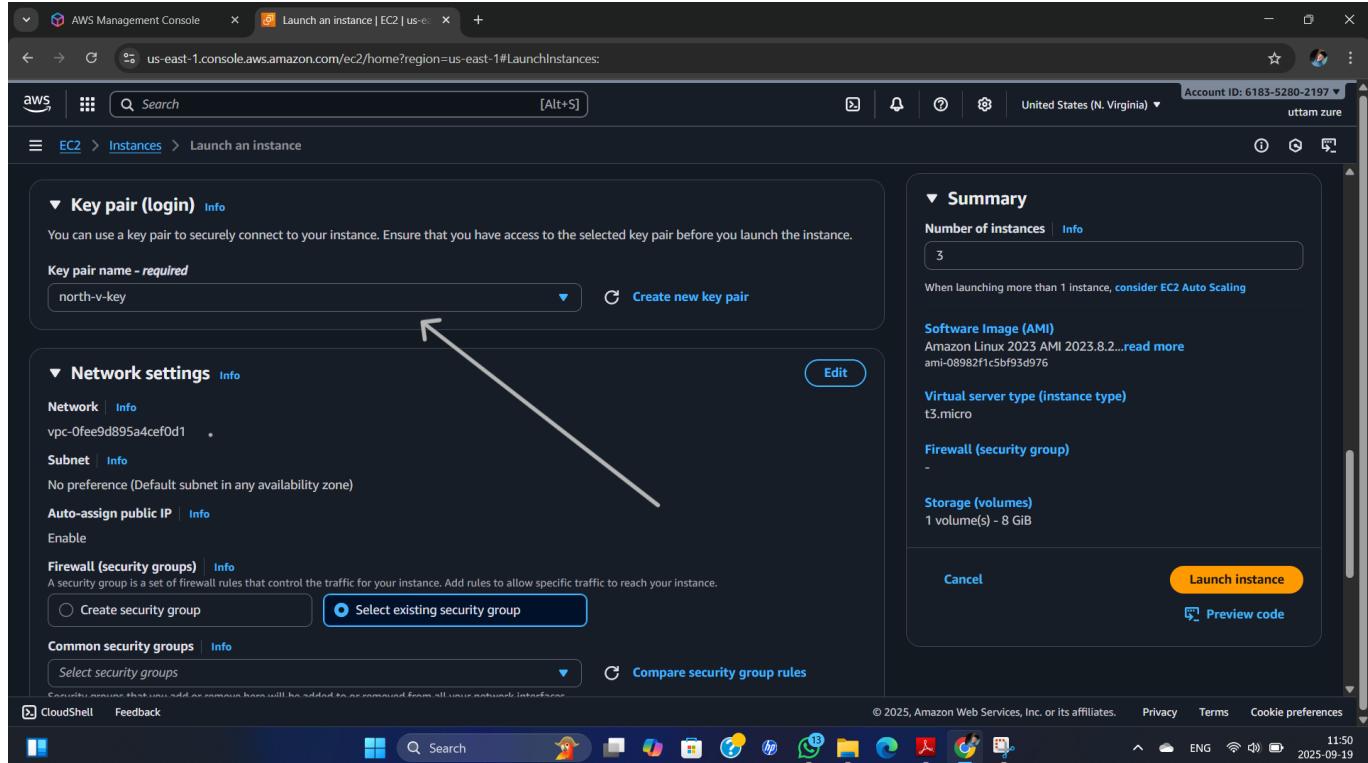
Steps for Implement

Step-1: Launch The Three Instance Server-1,Server-2,Server-3.🚀

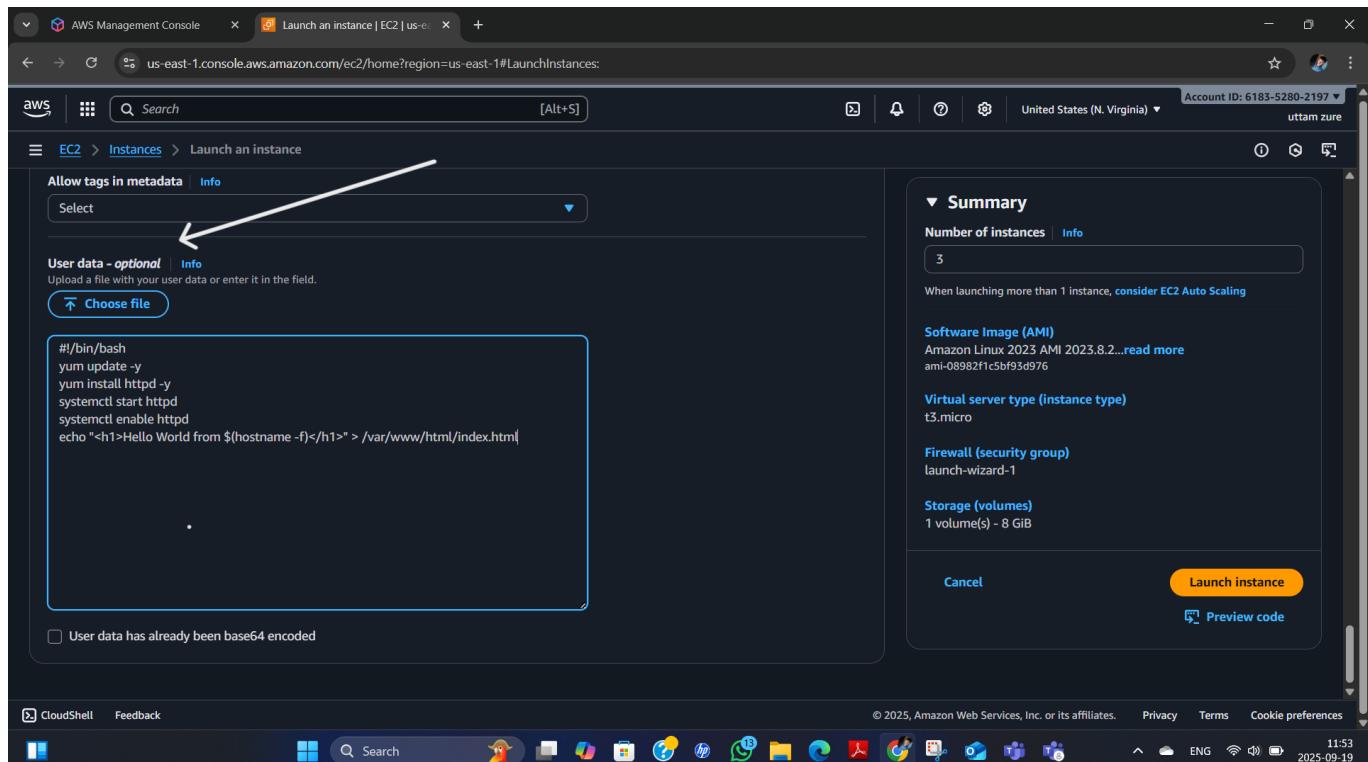
- Click on the lauch instance.
- Create a instances Server-1.

The screenshot shows the AWS Management Console interface for launching EC2 instances. The current step is 'Name and tags'. A blue arrow points to the 'Name' input field, which contains the value 'Server'. To the right, the 'Summary' section displays the configuration for launching three instances. It includes the selected AMI (Amazon Linux 2023 AMI 2023.8.2...), the instance type (t3.micro), and storage settings (1 volume(s) - 8 GiB). At the bottom right of the summary section is a prominent orange 'Launch instance' button.

- Select the key pair.



- Click on the advanced details.
- Go to user data optionl.



- Number of instance Three is creating the 3 instance.
- Click on the lanuch the instances.
- Edit the name of the server to server-1,server-2,server-3

The screenshot shows the AWS Management Console interface for the EC2 service. The left sidebar is collapsed, and the main area displays the 'Instances' section. A search bar at the top allows filtering by instance attribute or tag. Below it, a table lists three instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
Server-1	i-0f5eb6161367fd316	Running	t3.micro	Initializing	View alarms	us-east-1d	ec2-44-2
Server-2	i-0154c7af1efaffcf0	Running	t3.micro	Initializing	View alarms	us-east-1d	ec2-54-2
Server-3	i-0848eb999015174ff	Running	t3.micro	Initializing	View alarms	us-east-1d	ec2-23-2

Below the table, a message indicates '3 instances selected'. The 'Monitoring' tab is active, showing various metrics like CPU utilization, Network in, Network out, and Network packets. A 'Configure CloudWatch agent' button is available. The bottom of the screen shows the AWS navigation bar and system tray.

Step-2: Create the load balancer.

The screenshot shows the AWS Management Console interface for the Load Balancers service. The left sidebar is collapsed, and the main area displays the 'Load balancers' section. A search bar at the top allows filtering by load balancer name, state, type, scheme, IP address type, VPC ID, and availability zone. Below it, a message states 'No load balancers' and 'You don't have any load balancers in us-east-1'. A large blue 'Create load balancer' button is prominently displayed. The bottom of the screen shows the AWS navigation bar and system tray.

Step-3: Create the classic load balancer & create button.

Containers. **Create**

Classic Load Balancer Info

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.

Create

Close

Step-4: Assign the name to the classic load balancer

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

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.

Network mapping Info
The load balancer routes traffic to targets in the selected subnets, and in accordance with your network settings.

Step-5: Right click on the availability zones & subnets &Select the security group.

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

VPC | Info
loadBalancers.vpcDescriptionClbInternetFacing Learn more ↗

vpc-0fee9d895a4cef0d1 (default) Create VPC ↗
172.31.0.0/16

Availability Zones and subnets
Select at least one Availability Zone and one subnet for each zone. We recommend selecting at least two Availability Zones. The load balancer will route traffic only to targets in the selected Availability Zones. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

- us-east-1a (use1-az6)
- us-east-1b (use1-az1)
- us-east-1c (use1-az2)
- us-east-1d (use1-az4)
- us-east-1e (use1-az3)
- us-east-1f (use1-az5)

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 © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
Rainy days ahead 26°C 12:05 2025-09-19

Step-6: Add The Instances & Available Instances.

Instances (0)

You can add instances to register as targets of the load balancer. Alternatively, after your load balancer is created, you can add it to an Amazon EC2 Auto Scaling group to ensure you maintain the correct number of instances to handle the load for your application. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone.

Add instances

Attributes

Creating your load balancer using the console gives you the opportunity to specify additional features at launch. You can also find and adjust these settings in the load balancer's "Attributes" section after your load balancer is created.

- Enable cross-zone load balancing**
With cross-zone load balancing, each load balancer node for your Classic Load Balancer distributes requests evenly across the registered instances in all enabled Availability Zones. If cross-zone load balancing is disabled, each load balancer node distributes requests evenly across the registered instances in its Availability Zone only. Classic Load Balancers created with the API or CLI have cross-zone load balancing disabled by default. After you create a Classic Load Balancer, you can enable or disable cross-zone load balancing at any time.
- Enable connection draining**
Applicable to instances that are deregistering, this feature allows existing connections to complete (during a specified draining interval) before reporting the instance as deregistered. [Learn more](#)

Timeout (draining interval)
The maximum time for the load balancer to allow existing connections to complete. When the maximum time limit is reached, the load balancer forcibly closes any remaining connections and reports the instance as deregistered.

Ping protocol | **Ping port** | **Ping path**

Add instances

Select EC2 instances to register to your load balancer. Requests will be routed to registered instances that meet the health check requirements. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone enabled for the load balancer. If demand on your instances changes, you can register or deregister instances without disrupting the flow of requests to your application. [Learn more](#)

VPC
vpc-0fee9d895a4cef0d1

Available instances (3/3)

Instance ID	Name	State	Security groups	Zone	Public IPv4
i-0f5eb6161367fd316	Server-1	Running	launch-wizard-1	us-east-1d	44.223.23.4
i-0154c7af1efaffcf0	Server-2	Running	launch-wizard-1	us-east-1d	54.234.92.20
i-084eb999015174ff	Server-3	Running	launch-wizard-1	us-east-1d	23.20.24.220

Cancel | **Confirm**

CloudShell | **Feedback**

Step-7: click on the create loadbalancer.

Basic configuration

Name: Classic-LB
Scheme: Internet-facing

Network mapping

VPC: [vpc-0fee0d8b95a4cef0d1](#)
Availability Zones and subnets:

- us-east-1a [subnet-074a9b3cdf93b9296](#)
- us-east-1b [subnet-07be8c5f0c7468a8](#)
- us-east-1c [subnet-0aa3f695340632b60](#)
- us-east-1d [subnet-0bbaf99a3d66bf320](#)
- us-east-1e [subnet-040e9a97d1c583631](#)
- us-east-1f [subnet-0dcf17a856b497f30](#)

Health checks

HTTP:80/index.html

- Timeout: 2 seconds
- Interval: 5 seconds
- Unhealthy threshold: 2
- Healthy threshold: 10

Instances

3 instances added

Attributes

- Cross-zone load balancing: On
- Connection draining: On
- Connection draining timeout: 300 seconds

Tags

Create load balancer

Step-8: Copy the DNS name of the CLB.

Scheme: Internet-facing

HOSTED zone: Z355XDTRQ7X7K

Availability zones

- [subnet-07be8c5f0c7468a8](#) us-east-1b (use1-az1)
- [subnet-0aa3f695340632b60](#) us-east-1c (use1-az2)
- [subnet-074a9b3cdf93b9296](#) us-east-1a (use1-az6)
- [subnet-0bbaf99a3d66bf320](#) us-east-1d (use1-az4)
- [subnet-0dcf17a856b497f30](#) us-east-1f (use1-az5)
- [subnet-040e9a97d1c583631](#) us-east-1e (use1-az3)

DNS name info

Classic-LB-457716541.us-east-1.elb.amazonaws.com (A Record)

This Classic Load Balancer can be migrated to a next generation load balancer. Migration wizard uses your load balancer's current configurations to create a new load balancer. [Learn more](#)

Distribution of targets by Availability Zone (AZ)

For each enabled Availability Zone, you can view the number of registered instances and their current health states. Selecting any values here will apply the corresponding filter to the Target instances table.

Listeners | **Network mapping** | **Security** | **Health checks** | **Target instances** | **Monitoring** | **Attributes** | **Tags**

Step-8:

Expected Output

- Paste CLB DNS name in the new incognito window.

- **Server-1**



- **Server-2**



- **Server-3**



🔴 Summary (Classic Load Balancer) :-

The Classic Load Balancer (CLB) is one of AWS's legacy load balancing services. It operates at both Layer 4 (TCP) and Layer 7 (HTTP/HTTPS), distributing incoming traffic across multiple EC2 instances to improve application availability and reliability. CLB supports basic features such as SSL termination and sticky sessions, making it suitable for small to medium-scale applications. However, for modern and advanced workloads, AWS recommends using Application Load Balancer (ALB) or Network Load Balancer (NLB).