

Classic Load Balancer Setup and Configuration on AWS

Introduction

This project demonstrates the deployment and configuration of a Classic Load Balancer (CLB) in AWS. A Classic Load Balancer is designed to automatically distribute incoming traffic across multiple EC2 instances, ensuring improved availability, fault tolerance, and scalability of applications. The Classic Load Balancer operates at both the transport layer (Layer 4) and the application layer (Layer 7), giving it the ability to balance traffic based on both network connections and application requests. It continuously performs health checks to route traffic only to healthy instances, further enhancing fault tolerance.

Prerequisites

Before setting up the Classic Load Balancer, ensure the following requirements are met:

- AWS Account – with access to EC2 and Load Balancer services
- Running EC2 Instances – at least two instances in the same VPC and region
- Security Groups configured to allow:
 - Inbound HTTP (port 80) and/or HTTPS (port 443) traffic
 - Inbound SSH (port 22) access for administration (optional)
- Web Server Installed (e.g., Apache or Nginx) on each EC2 instance, serving a sample page

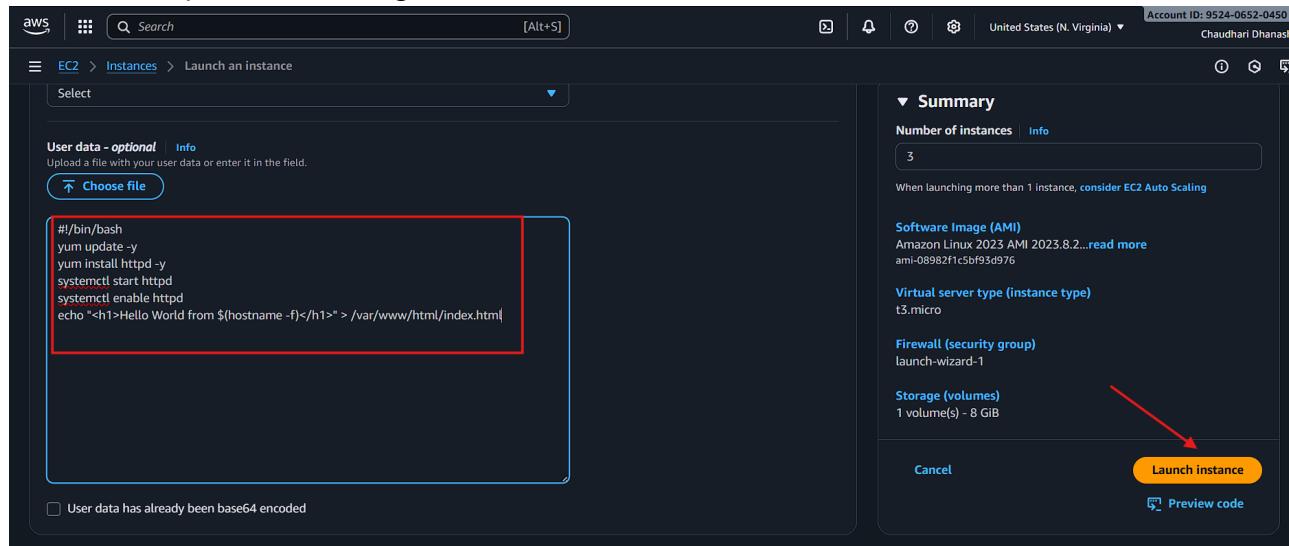
Steps to Setup Classic Load Balancer

Step 1: Launch 3 EC2 Instances with User Data Script

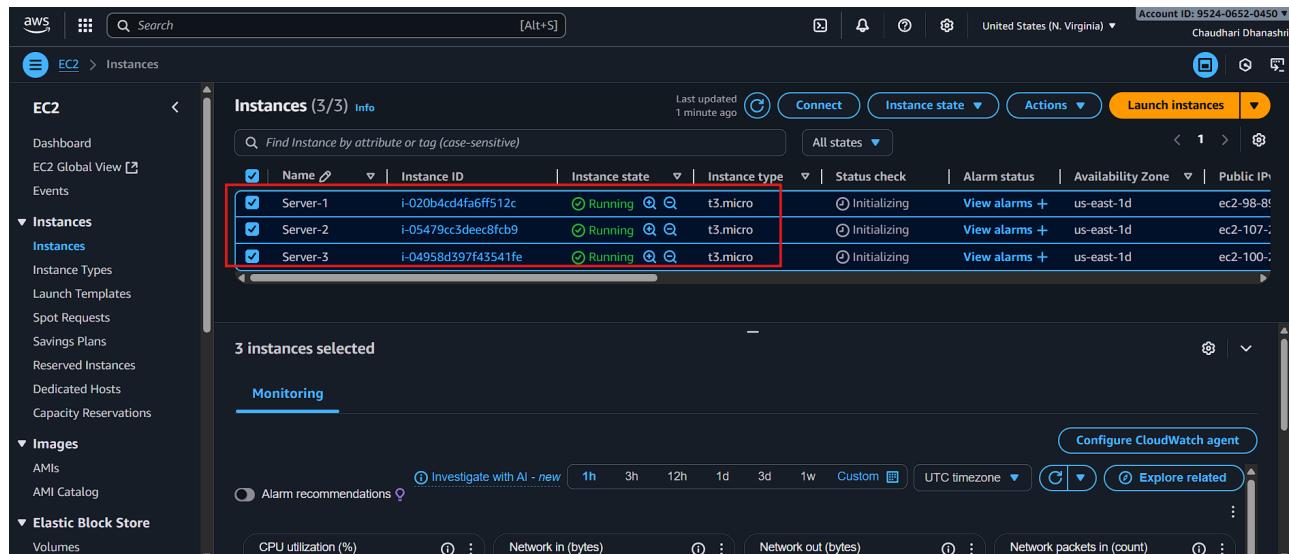
1. Launch 3 Instance.

The screenshot shows the AWS EC2 'Launch an instance' wizard. The 'Number of instances' field is highlighted with a red box and contains the value '3'. Other visible fields include 'Name' (set to 'Server'), 'Software Image (AMI)' (set to 'Amazon Linux 2023.8.2...'), 'Virtual server type (Instance type)' (set to 't3.micro'), and 'Storage (volumes)' (set to '1 volume(s) - 8 GiB'). The 'Summary' section on the right lists these configurations. At the bottom, there are tabs for 'Recents' and 'Quick Start'.

2. Write user script while launching instance.

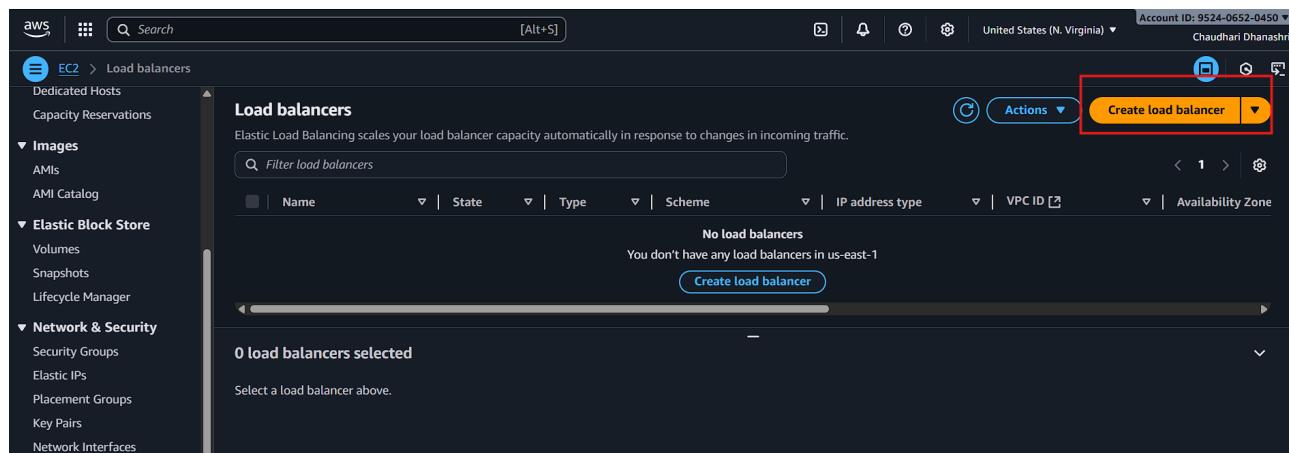


3. Rename the 3 instance with different names(server-1, server-2, server-3)



Step 2: Create a Classic Load Balancer

1. Go to Load Balancer.



2. Choose Classic Load Balancer.

The screenshot shows the AWS EC2 Load Balancers page. In the top navigation bar, 'EC2' and 'Load balancers' are selected. Below this, there are two main sections: 'Containers' and 'Classic Load Balancer - previous generation'. The 'Classic Load Balancer' section contains a diagram illustrating its architecture: a client connects to a CLB (Classic Load Balancer) which then routes traffic to multiple EC2 instances via HTTP, HTTPS, TCP, and SSL protocols. A red box highlights the 'Create' button in this section.

3. Name the Load Balancer

The screenshot shows the 'Create Classic Load Balancer' wizard. The first step, 'Basic configuration', is displayed. It includes fields for 'Load balancer name' (containing 'classic-LB'), 'Scheme' (set to 'Internet-facing'), and 'Internal' options. A red box highlights the 'classic-LB' input field.

4. Select all Availability Zones.

The screenshot shows the 'Create Classic Load Balancer' wizard at the 'Availability Zones and subnets' step. Three availability zones are selected: 'us-east-1a (use1-az6)', 'us-east-1b (use1-az1)', and 'us-east-1c (use1-az2)'. Each selection is highlighted with a red box. The subnets for each zone are listed below, showing their respective CIDRs.

Subnet	CIDR
subnet-0074946e1e6cb0a0a	IPv4 subnet CIDR: 172.31.32.0/20
subnet-0aa2af906dbf510a7	IPv4 subnet CIDR: 172.31.0.0/20
subnet-012dbf3ca55b1eabb	IPv4 subnet CIDR: 172.31.80.0/20

5. Manage Security Group

The screenshot shows the AWS Security Groups page. A security group named "launch-wizard-1" is selected, highlighted with a red box. The VPC is listed as "vpc-0d81f010b0ccfe3d1".

6. Add instances to Load Balancer

The screenshot shows the "Create Classic Load Balancer" page under the EC2 > Load balancers section. In the "Instances (0)" section, there is a table header and a message stating "No instances added". A red box highlights the "Add instances" button.

The screenshot shows the "Add instances" modal. It lists three available instances: "Server-1", "Server-2", and "Server-3", all of which are checked and highlighted with a red box. The modal includes a "Confirm" button at the bottom right.

The screenshot shows the "Create Classic Load Balancer" page with the following configuration details:

- Scheme:** Internet-facing
- Availability Zones and subnets:**
 - us-east-1a: subnet-0074946e1e6cb0a0a
 - us-east-1b: subnet-0a0a2f906dbf510a7
 - us-east-1c: subnet-012dbf3ca55b1eabb
 - us-east-1d: subnet-067179291e79afecf
 - us-east-1e: subnet-005b34b347ce0ae30
 - us-east-1f: subnet-0b0efb4855b5495f6
- Health checks:** HTTP:80/index.html
 - Timeout: 2 seconds
 - Interval: 5 seconds
 - Unhealthy threshold: 2
 - Healthy threshold: 10
- Instances:** 3 instances added
 - 3 instances in us-east-1d
- Attributes:**
 - Cross-zone load balancing: On
 - Connection draining: On
 - Connection draining timeout: 300 seconds
- Tags:** -

A red box highlights the "Create load balancer" button at the bottom right.

7. Copy DNS command and paste it in any browser.

The screenshot shows the AWS EC2 Load Balancers console. On the left, there's a navigation sidebar with sections like Images, Elastic Block Store, Network & Security, and Load Balancing. Under Load Balancing, 'Load Balancers' is selected. The main area displays a table titled 'Load balancers (1/1)'. A single row is shown for a 'classic-LB' load balancer, which is highlighted with a red box. Below the table, a section titled 'Load balancer: classic-LB' shows the 'DNS name info' for the load balancer. The DNS name 'classic-LB-1630018076.us-east-1.elb.amazonaws.com (A Record)' is also highlighted with a red box. A tooltip indicates that this load balancer can be migrated to a next-generation load balancer using the current configurations.

Step 3: Testing the Deployment.

1. Output for Server-1

The screenshot shows a web browser window with the URL 'classic-lb-1630018076.us-east-1.elb.amazonaws.com'. The page content is 'Hello World from ip-172-31-30-243.ec2.internal', indicating that the load balancer is correctly routing traffic to the intended server.

2. Output for Server-2



3. Output for Server-3



Summary

This project demonstrates the deployment and configuration of a Classic Load Balancer (CLB) in AWS to distribute incoming traffic across multiple EC2 instances. By launching three instances with a user-data script, the setup ensures that each instance runs a web server and serves unique content, allowing you to verify traffic distribution.