



TrainerTests
.com

This lab demonstrates the steps from *Demo: Create a Load Balancer*.

My full AWS Architect Associate course can be found here:

<https://www.udemy.com/course/ultimateaws/?referralCode=7ED214B795C444141361>

Lab Guide: Setting Up a Network Load Balancer in AWS

This lab guide walks you through the process of setting up a **Network Load Balancer (NLB)** in AWS to distribute traffic across multiple EC2 instances in different Availability Zones. By following these steps, you will gain hands-on experience configuring and testing an NLB while understanding its key concepts.

Lab Objectives

1. Launch two EC2 instances in different Availability Zones.
 2. Configure a target group to include the EC2 instances.
 3. Create and associate a Network Load Balancer with the target group.
 4. Test the load balancer's functionality and observe how it handles instance failures.
-

Step 1: Launch Two EC2 Instances

1.1 Log into the AWS Console

1. Log into your AWS Console and ensure you're in the **Ohio region (us-east-2)**.
2. Navigate to the **EC2 Dashboard**.

1.2 Launch the First EC2 Instance

1. Click **Launch Instance** and follow these settings:
 - o **Name:** webserver01a.
 - o **AMI:** Amazon Linux 2 AMI.
 - o **Instance Type:** t2.micro (or another free-tier eligible type).

- **Availability Zone:** Ensure it is in **us-east-2b**.
 - **Key Pair:** Select or create a key pair for SSH access.
 - **Security Group:**
 - Create a new security group checking **Allow HTTPS traffic from the internet** and **Allow HTTP traffic from the internet**.
 - Click **Advanced details** and scroll down to **User data – optional**.
 - Run the **User Data Script** included in the course resources
2. Launch the instance.

1.3 Launch the Second EC2 Instance

1. Repeat the steps above with the following changes:
 - **Name:** `Webserver02a`.
 - **Availability Zone:** Ensure it is in **us-east-2a**.
 - Use the **same security group** and **key pair** as the first instance.
 - Use the same user data script.
 2. Launch the instance.
-

Step 2: Create a Target Group

2.1 Navigate to Target Groups

1. In the **EC2 Dashboard**, scroll down the left menu and select **Target Groups** under **Load Balancing**.

2.2 Create the Target Group

1. Click **Create Target Group** and configure as follows:
 - **Target Type:** EC2 Instances.
 - **Name:** `DemoTargetGroup`.
 - **Protocol:** HTTP.
 - **Port:** 80.
 - **Health Check Path:** `/`.
 - Leave other settings as default.
2. Click **Next**.

2.3 Register Targets

1. Select both EC2 instances (`Webserver01a` and `Webserver02a`) and include them in the target group.
 2. Click **Create Target Group**.
-

Step 3: Create a Network Load Balancer

3.1 Navigate to Load Balancers

1. In the **EC2 Dashboard**, scroll to **Load Balancers** under **Load Balancing**.

3.2 Configure the Load Balancer

1. Click **Create Load Balancer** and select **Network Load Balancer**.
2. Set the following:
 - **Name:** DemoNLB.
 - **Scheme:** Internet-facing.
 - **Availability Zones:** Select all AZs and ensure `us-east-2a` and `us-east-2b` are included.
 - **Security Group:** `WebServers` (not default). It must allow HTTP traffic.
 - **Listeners:**
 - Protocol: HTTP, Port: 80.
3. Click **Next**.

3.3 Associate the Target Group

1. Under **Listeners and Routing**, select `DemoTargetGroup` as the target group.
 2. Click **Create Load Balancer**.
-

Step 4: Test the Network Load Balancer

4.1 Retrieve the Load Balancer's DNS

1. Go to **Load Balancers** in the EC2 Dashboard.
2. Select `DemoNLB` and copy its **DNS Name**.

4.2 Test the Load Balancer

1. Open a web browser and paste the DNS name.
 2. Refresh the page multiple times to observe the traffic being distributed between `Webserver01a` and `Webserver02a`.
 - The browser should alternate between displaying "Webserver01a" and "Webserver02a."
-

Step 5: Simulate an Instance Failure

5.1 Terminate One EC2 Instance

1. Go to **Instances** in the EC2 Dashboard.
2. Select `Webserver02a` and terminate it.

5.2 Verify Target Group Health

1. Return to **Target Groups** and select `DemoTargetGroup`.
2. Go to the **Targets** tab and observe the health status.
 - The terminated instance will show as **unhealthy** and eventually be removed from the target group.
3. Test the load balancer's DNS again. It should now only serve traffic from `Webserver01a`.

Key Concepts

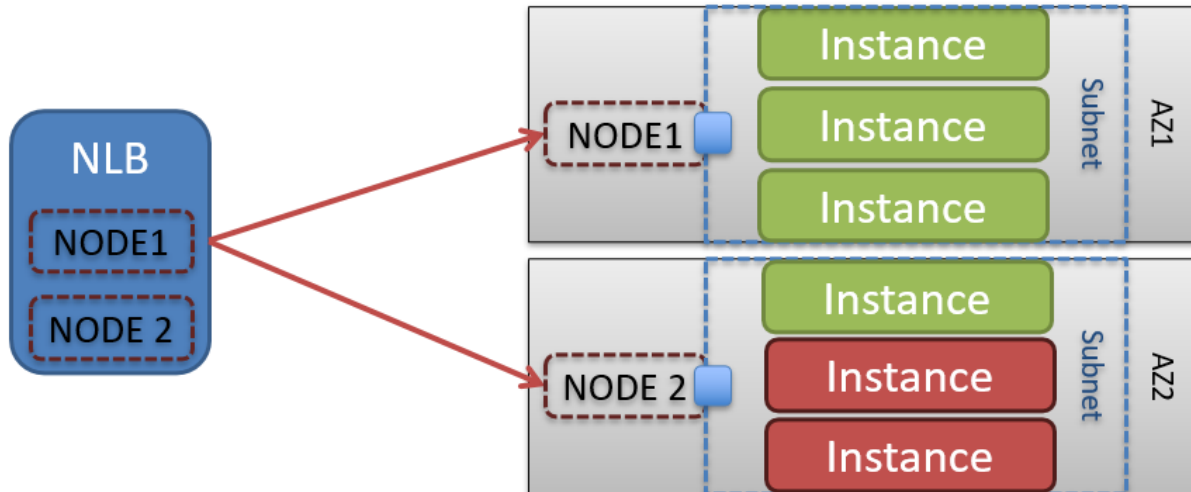
1. **Network Load Balancer (NLB):**
 - Distributes traffic across multiple targets, such as EC2 instances, for high availability and scalability.
 - Operates at the transport layer (Layer 4).
2. **Target Groups:**
 - A collection of resources (EC2 instances, IPs, or Lambda functions) that receive traffic from the load balancer.
3. **Health Checks:**
 - Periodic checks to determine the availability of targets. Traffic is only routed to healthy targets.
4. **Multi-AZ Deployment:**
 - Distributing targets across multiple Availability Zones ensures high availability even if one AZ fails.
5. **User Data Scripts:**
 - Automates instance setup tasks such as installing software and configuring web servers.

Summary

In this lab, you set up a Network Load Balancer to distribute traffic across EC2 instances in multiple Availability Zones. You also learned how the load balancer detects and handles instance failures using health checks, ensuring continued application availability. By completing this lab, you now understand the fundamentals of configuring and testing an NLB in AWS.

See slide below:

Network Load Balancer Example



For more details see my full AWS Architect Associate course:

<https://www.udemy.com/course/ultimateaws/?referralCode=7ED214B795C444141361>