**EX: 05 NAME: M. NIVASHINI**

**DATE: 30.07.2024 ROLL.NO.: 23MCA020**

**Load Balancing Traffic**

**Aim:**

The aim is to set up a free-tier Application Load Balancer to efficiently distribute traffic across multiple free-tier EC2 instances, ensuring balanced load and high availability. Additionally, test and experiment with different load balancing algorithms to optimize performance.

**Theoretical Background**

**1. Set up a Load Balancer**

**1.1. Log in to AWS Management Console**

1. **Sign in** to the AWS Management Console.

**1.2. Open the EC2 Dashboard**

1. **Navigate** to **Services** and select **EC2**.

**1.3. Create a Free-Tier Application Load Balancer**

1. **Open Load Balancers:**
   * In the EC2 Dashboard, scroll down to **Load Balancing** and select **Load Balancers**.
2. **Create Load Balancer:**
   * Click **Create Load Balancer**.
   * Select **Application Load Balancer** (free-tier eligible).
   * Click **Create**.
3. **Configure Load Balancer:**
   * **Name:** Enter a name for your load balancer (e.g., MyFreeTierLoadBalancer).
   * **Scheme:** Choose **Internet-facing** (for public access).
   * **IP Address Type:** Choose **IPv4**.
   * Click **Next: Configure Security Settings**.
4. **Configure Security Settings:**
   * **For HTTP:** No SSL certificate needed for HTTP.
   * Click **Next: Configure Security Groups**.
5. **Configure Security Groups:**
   * Select an existing security group or create a new one that allows traffic on HTTP (port 80).
   * Click **Next: Configure Routing**.
6. **Configure Routing:**
   * **Target Group:** Create a new target group.
     + **Target Type:** Choose **Instance**.
     + **Protocol:** HTTP.
     + **Port:** 80.
   * Click **Next: Register Targets**.
7. **Register Targets:**
   * You can skip this step for now and register targets later if no instances are ready.
   * Click **Next: Review**.
8. **Review and Create:**
   * Review the settings and click **Create Load Balancer**.

**2. Deploy Applications on Free-Tier EC2 Instances**

**2.1. Launch Free-Tier EC2 Instances**

1. **Open the EC2 Dashboard:**
   * Click **Launch Instance**.
2. **Choose an Amazon Machine Image (AMI):**
   * Select a free-tier eligible AMI like Amazon Linux 2 or Ubuntu Server.
3. **Choose an Instance Type:**
   * Select t2.micro (free-tier eligible).
4. **Configure Instance Details:**
   * Configure as needed, then click **Next: Add Storage**.
5. **Add Storage:**
   * Use default settings and click **Next: Add Tags**.
6. **Add Tags:**
   * Add tags (e.g., Name: MyAppInstance), then click **Next: Configure Security Group**.
7. **Configure Security Group:**
   * Create a new security group allowing HTTP (port 80).
   * Click **Review and Launch**.
8. **Review and Launch:**
   * Review settings and click **Launch**, select or create a key pair, then click **Launch Instances**.

**2.2. Deploy Applications**

1. **Access EC2 Instances:**
   * SSH into Linux instances or use the appropriate method for application deployment.
2. **Deploy Application:**
   * Install a web server (e.g., Apache or Nginx) and deploy a simple web application.

**3. Test Load Balancer**

**3.1. Obtain Load Balancer DNS Name**

1. **Open Load Balancers:**
   * In the EC2 Dashboard, select **Load Balancers**.
2. **Copy DNS Name:**
   * Find your load balancer and copy its DNS name.

**3.2. Test Traffic Distribution**

1. **Open a Web Browser:**
   * Enter the load balancer’s DNS name.
2. **Verify Application:**
   * Ensure that the application is accessible and check if requests are distributed across instances.

**4. Experiment with Load Balancing Algorithms**

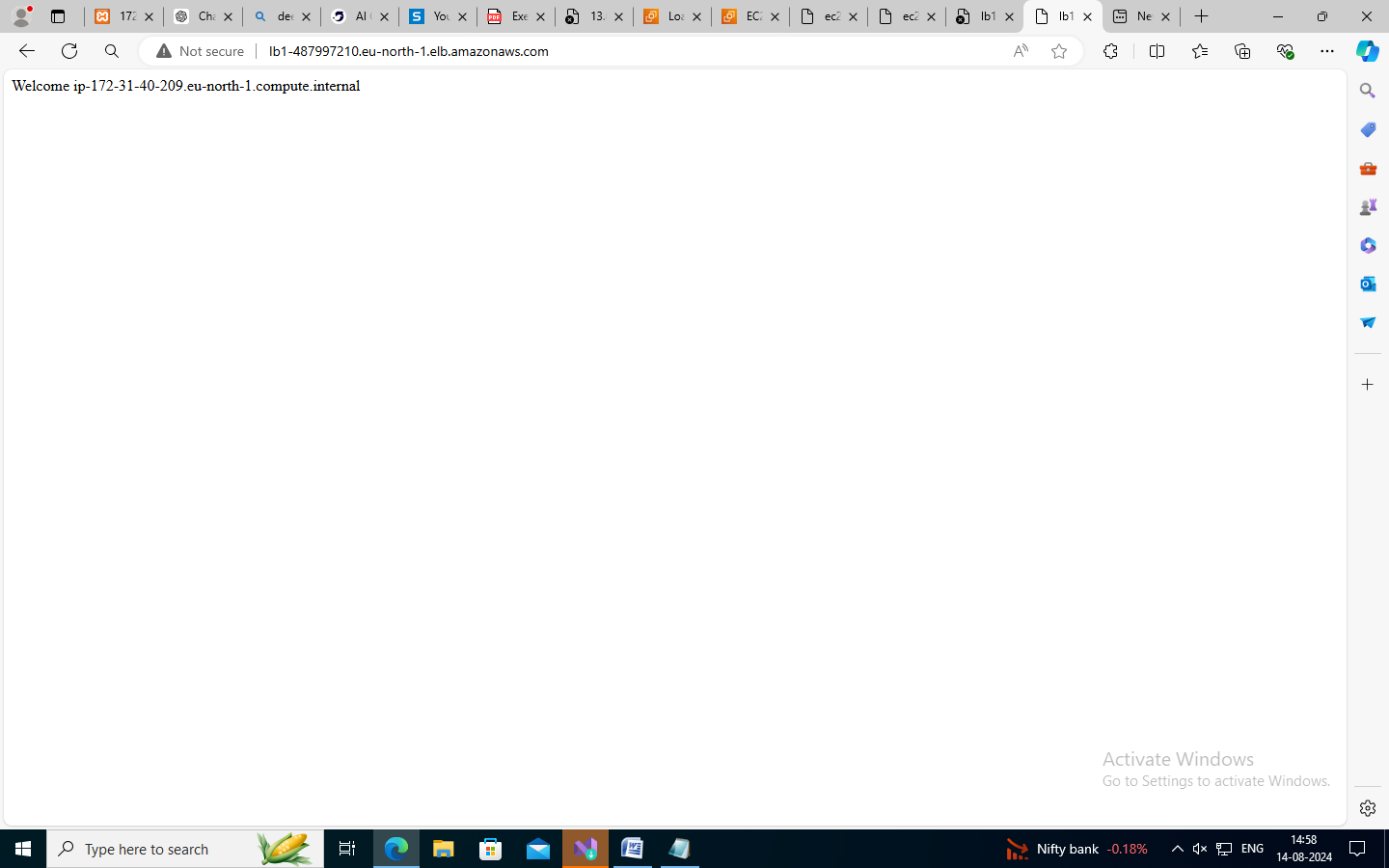
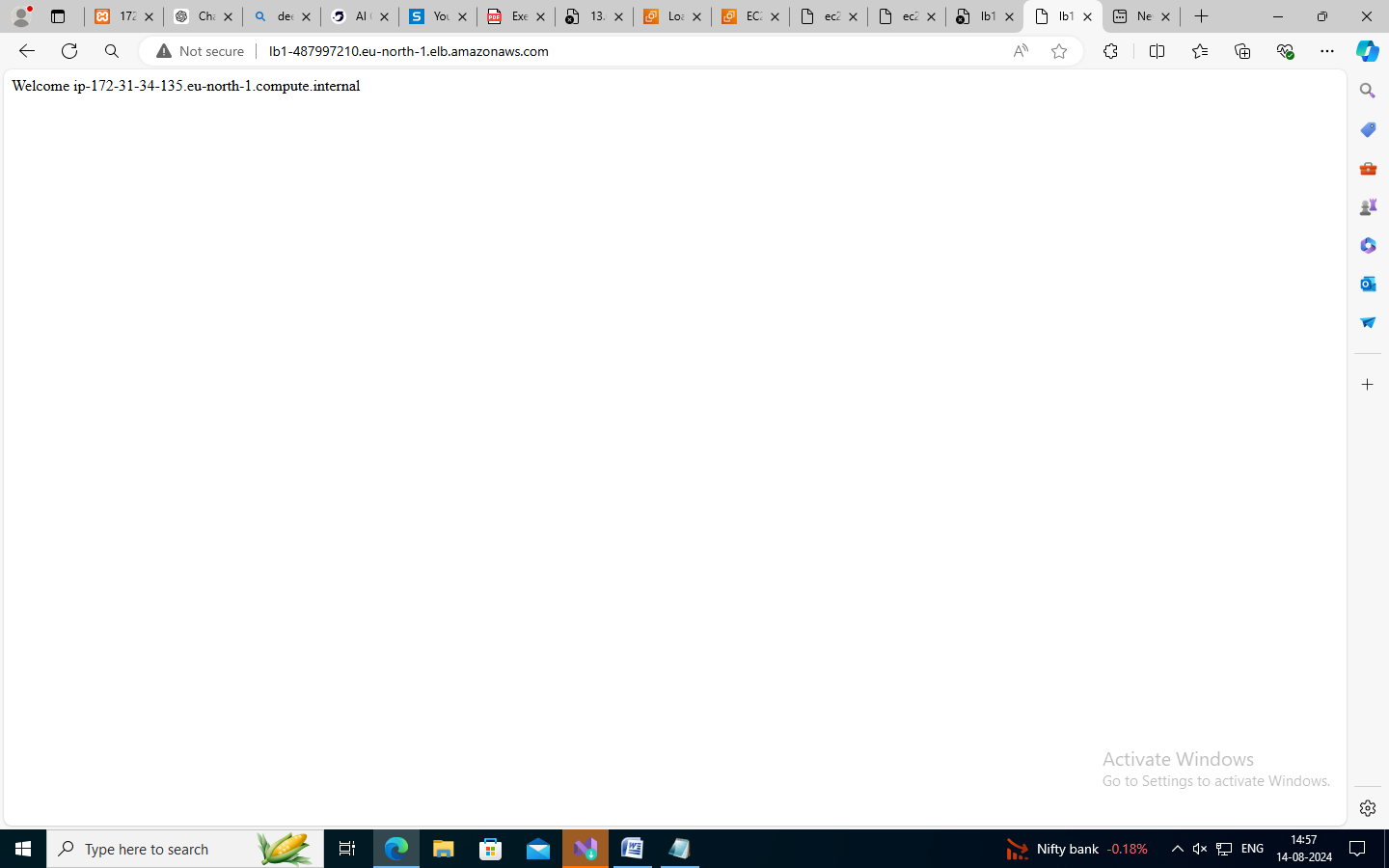
**4.1. Modify Load Balancer Algorithm**

1. **Modify Target Group:**
   * In the EC2 Dashboard, go to **Target Groups**.
   * Select your target group.
2. **View and Change Algorithm:**
   * For Application Load Balancer, the default algorithm is round-robin. Check **Attributes** or **Advanced** settings if available.

**4.2. Observe Different Algorithms**

1. **Test Traffic Distribution:**
   * Access the application through the load balancer and observe how traffic distribution changes with different algorithms.

**Output**

****

**Conclusion**

Thus the creation and managing Windows and Linux EC2 instances, added and attached new volumes, and took snapshots for backup or replication. Additionally, we resized volumes and verified changes, ensuring effective storage management across different operating systems has been developed and executed successfully.