Cloud Computing Exercise #19

Load Balancing

A. Preparation

1. Sign in to your AWS account as the non-root admin user.

B. Create and launch two EC2 instances

1. Go to EC2 Security groups (EC2/Network & security/Security groups) and create a security group for your instances called “MySecurityGroup” that allows SSH (port 22) and HTTP (port 80) inbound traffic from anywhere and allows all outbound traffic.
2. Go to EC2 (EC2/Instances/Instances) and launch two EC2 instances. Select the “Amazon Linux 2023 AMI” from Amazon Marketplace as the AMI, and t2.micro as the instance type. On the “Configure Instance Details” page, select the subnet that belongs in the “us-east-1a” AZ for the first EC2 instance, and select the subnet that belongs to the “us-east-1b” AZ for the second instance. (That is, the two EC2 instances should be launched in two separate AZs.) Choose your existing key pair (e.g. “mykey”) for instance access, and set the security group to the security group you created in step 2 (MySecurityGroup).
3. Install the nano text editor (package name: nano) and the Apache Web Server (package name: httpd) on both of the instances. (sudo yum install -y nano httpd)
4. Go to the directory /var/www/html and write the command sudo nano index.html. You have to create an index.html file on both instances. Go to the directory /var/www/html and launch nano using sudo. Then copy and paste the following html code into nano’s text editor window:

<html>

<head>

<title>Welcome to my website!</title>

</head>

<body>

<h1>Greetings from Server#$$number$$ with IP address $$private IP$$!</h1>

</body>

</html>

For the first instance, replace $$number$$ with 1, and for the second instance, replace it with 2. For both instances, replace $$private IP$$ with the private IP address of that instance.

1. Start the Apache web server in both instances (sudo systemctl start httpd). Check the status of the server process and verify that it is running (sudo systemctl status httpd). Test the web servers by entering their public IP addresses in your web browser and checking that you see the greetings message from both Server#1 and Server#2. Now we can access both servers directly, but there is no load balancing across these two instances yet.

C. Create an Elastic Load Balancer

1. Go to the load balancer dashboard (EC2/Load Balancing/Load balancers) and create a new load balancer of type “Application Load Balancer”. Give it a name (e.g. “MyLoadBalancer”), and configure it as an Internet-facing load balancer with the IPv4 protocol (default values). In the “Network Mapping” section, select two AZs: us-east-1a and us-east-1b (the AZs where your EC2 instances are), and in the “Security Groups” section, select the security group you created previously (MySecurityGroup).
2. In the “Listeners and routing” section, select “Create target group” and the target type should be “Instances” (default value). Give the target group a name (e.g. “MyTargetGroup”), set the protocol to HTTP (port 80) and the protocol version to HTTP/1.1 (default values). In the “Health Checks” section, set the health check protocol to HTTP and provide the path “/index.html” as the health check path. Click “Next”, select your two running EC2 instances and add them to the target group by **selecting “Include as pending below”.** Finally select “Create target group”. Your new target group should appear on the list of target groups at EC2/Target groups. If you click on your target group, you should see the configuration details, including the two EC2 instances in the list of registered targets.
3. Go back to the load balancer creation page, and in the “Listeners and routing” section, select the target group you created (MyTargetGroup). Scroll down all the way, and create the load balancer. In the list of your load balancers (EC2/Load balancing/Load balancer), the new load balancer should show up. Check its state: it should be “Provisioning” – wait until its state becomes “Active” (it may take a few minutes).

D. Test your load balancer

1. Below the list of the load balancers, you can find the detail page for the selected load balancer. Select the “Description” tab, and find the DNS name (domain name) for the load balancer. Copy the URL to your browser window’s address bar, and you should see one of the web server’s greeting messages. If you reload the web page multiple times, you should see that the greeting messages are alternating between those of Server#1 and Server#2, showing that the requests are forwarded to the two EC2 instances in a round robin fashion.
2. Terminate one of your instances manually, but keep the other instance running. If you reload the pages multiple times again, you will see that even though one of the web server instances are down, the overall service is still available – the load balancer keeps serving the web page from the EC2 instance that is still running (Server#1 or Server#2).

F. Clean up after yourself

1. Go to EC2 (EC2/Instances/Instances) and terminate the still running EC2 instance. Go to EBS (EC2/Elastic Block Store/Volumes) and delete any EBS volumes that may have survived the EC2 instance termination. Go to the load balancers (EC2/Load balancing/Load balancers) and delete your load balancer. Go to the target groups (EC2/Load balancing/Target groups) and delete your target group.
2. Log out of AWS.