Day8 of learning AWS

13 February 2025 09:05 AM

Elastic Load Balancer

What is LB?

Load Balancer (LB) is like a traffic manager for your web applications. It evenly distributes incoming web traffic among multiple servers, ensuring no single server gets overwhelmed, thus improving performance and reliability.

What is Application Load Balancer?

An Application Load Balancer (ALB) is a type of LB that operates at the application layer, making it intelligent and capable of routing traffic based on content. It's specifically designed for modern applications that rely on multiple services.

Key Components of an Application Load Balancer

1. **Load Balancer:**
   - Serves as the primary contact point for clients.
   - Distributes incoming application traffic to various targets, like EC2 instances, across multiple Availability Zones, enhancing overall application availability.
2. **Listeners:**
   - Responsible for handling connection requests from clients.
   - Operate on specified protocols and ports that you configure.
   - Rules defined for each listener guide how the load balancer routes requests to its registered targets.
   - Rules include a priority, one or more actions, and conditions. Actions are executed when rule conditions are met.
   - It's crucial to define a default rule for each listener, and additional rules can be optionally configured.
3. **Target Groups:**
   - Route requests to one or more registered targets, such as EC2 instances.
   - Defined by the protocol and port number specified.
   - Targets can be registered with multiple target groups for added flexibility.
   - Health checks are configurable on a per target group basis.
   - Health checks assess the well-being of all targets within a group specified in a listener rule for the load balancer.

How Application Load Balancer Works:

- **Content-Based Routing:** ALB can route traffic based on content, examining HTTP/HTTPS headers, allowing for advanced routing decisions.
- **Path-Based Routing:** It can route requests based on the URL path, directing traffic to different backend services.
- **Host-Based Routing:** ALB supports routing based on the host header, enabling serving multiple applications from the same load balancer.
- **Containerized Applications:** ALB is well-suited for containerized applications, efficiently managing traffic within a containerized environment.

Why Application LB has More Functions Compared to Network LB:

- **Layer 7 vs. Layer 4:** Application LB operates at Layer 7 (application layer), offering more advanced routing based on content, while Network LB operates at Layer 4 (transport layer), focusing on routing based on IP addresses and ports.
- **Content Awareness:** ALB understands the content of the traffic, making it ideal for modern applications with dynamic routing needs.
- **Flexible Routing:** ALB's ability to route based on various factors like path, host, and content provides greater flexibility compared to Network LB.

Demo Link : AWS Tutorials - 39 - Elastic Load Balancer in AWS (ELB) - Classic Load Balancer(CLB)

Creating and Testing Your Load Balancer

This tutorial offers a practical introduction to Application Load Balancers using the AWS Management Console, an online interface. Follow these steps to create your initial Application Load Balancer.

Step 1: Configure Your Target Group

To set up a target group for managing requests, follow these simple steps:

1. Go to the Amazon EC2 console by clicking here.
2. In the navigation pane, find "Load Balancing" and click on "Target Groups."
3. Select "Create target group."
4. In the Basic configuration, leave the Target type as "instance."
5. Give your new target group a name in the "Target group name" field.
6. Keep the default settings for protocol (HTTP) and port (80).
7. Choose the VPC containing your instances and maintain the protocol version as HTTP1.
8. For Health checks, stick with the default settings and click "Next."
9. On the "Register targets" page, follow these optional steps. This step is not mandatory for creating the load balancer, but if you want to test your load balancer and ensure it routes traffic correctly, you should register this target.
   - Choose one or more instances from the "Available instances."
   - Keep the default port as 80 and select "Include as pending" below.
10. Click "Create target group" to finish.

Step 2: Choose a Load Balancer Type

Elastic Load Balancing provides various load balancers, and in this guide, we'll focus on creating an Application Load Balancer. Follow these steps to create an Application Load Balancer:

1. Visit the Amazon EC2 console by going to https://console.aws.amazon.com/ec2/.
2. On the navigation bar, select a Region for your load balancer. Make sure it's the same Region you used for your EC2 instances.
3. In the navigation pane under Load Balancing, click on Load Balancers.
4. Choose "Create Load Balancer."
5. Select "Create" for Application Load Balancer.

Step 3: Configure Your Load Balancer and Listener

To establish an Application Load Balancer, start by providing essential configuration details such as a name, scheme, and IP address type. Subsequently, input information regarding your network and configure one or more listeners. A listener acts as a process that monitors connection requests, configured with a specified protocol and port for client connections to the load balancer. Refer to the Listener configuration for details on supported protocols and ports.
Follow these steps to configure your load balancer and listener:

1. **Load Balancer Name:** Enter a name for your load balancer, e.g., "my-alb."
2. **Scheme and IP Address Type:** Retain the default values.
3. **Network Mapping:** Choose the VPC used for your EC2 instances. Select at least two Availability Zones and one subnet per zone. For each Availability Zone used to launch EC2 instances, select the Zone and choose one public subnet for that Zone.
4. **Security Groups:** Choose the default security group for the selected VPC. Alternatively, select a different security group. Ensure the security group has rules permitting communication between the load balancer and registered targets on both the listener port and the health check port. Refer to Security group rules for more details.
5. **Listeners and Routing:** Keep the default protocol and port. Select your target group from the list. This configures a listener accepting HTTP traffic on port 80, forwarding it to the chosen target group by default. No HTTPS listener is created for this tutorial.
6. **Default Action:** Choose the target group created and registered in Step 1: Configure your target group.
7. **Optional: Add a Tag:** Include a tag to categorize your load balancer. Tag keys must be unique, allowing letters, spaces, numbers (in UTF-8), and special characters like + - = . _ : / @. Avoid leading or trailing spaces, and note that tag values are case-sensitive.
8. **Review Configuration:** Assess your settings and select "Create load balancer." The system applies a few default attributes during creation, editable post-creation. Refer to Load balancer attributes for additional information.

Step 4: Verify Your Load Balancer's Performance

After creating the load balancer, ensure it's directing traffic to your EC2 instances.
Once you've successfully created the load balancer, ensure that it effectively directs traffic to your EC2 instances.
To check your load balancer, follow these steps:

1. **Close Notification:**
   - Once you receive confirmation that your load balancer was created successfully, click "Close."
2. **Navigate to Target Groups:**
   - In the navigation pane, go to "Load Balancing" and select "Target Groups."
3. **Select Target Group:**
   - Pick the newly created target group.
4. **Verify Instance Readiness:**
   - Click on "Targets" and confirm that your instances are ready. If an instance's status is "initial," it may still be in the registration process or has not passed the minimum health checks to be considered healthy. Once at least one instance shows a healthy status, you can proceed to test your load balancer.
5. **Access Load Balancers:**
   - In the navigation pane, under "Load Balancing," choose "Load Balancers."

6. **Choose Load Balancer:**
   - Select the recently created load balancer.
7. **Copy DNS Name:**
   - Under "Description," copy the DNS name of the load balancer (e.g., my-load-balancer-1234567890abcdef.elb.us-east-2.amazonaws.com).
8. **Paste in Browser:**
   - Paste the copied DNS name into the address field of an internet-connected web browser. If everything is functioning correctly, the browser should display the default page of your server.
9. **Optional - Add Listener Rules:**
   - To include additional listener rules, refer to "Add a rule." To enhance your setup, consider adding additional listener rules.

By following these steps, you've configured and tested your load balancer, ensuring efficient traffic distribution to your EC2 instances.

AWS Tutorials - 41 - Delete Classic Load Balancer