Table of contents

Introduction to AWS-ELB :

- Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones
- So to sum it up , ELBs are must for high availability.

ALB vs NLB (Ignoring classic LB being legacy)

- Network Load Balancer — This is the distribution of traffic based on network variables, such as IP address and destination ports. It is layer 4 (TCP) and below and is not designed to take into consideration anything at the application layer such as content type, cookie data, custom headers, user location, or the application behavior. .
- Application Load Balancer — This is the distribution of requests based on multiple variables, from the network layer to the application layer. It is context-aware and can direct requests based on any single variable as easily as it can a combination of variables. Applications are load balanced based on their peculiar behavior and not solely on server (operating system or virtualization layer) information.
- Application Load Balancer (as the name implies) works at the Application Layer (Layer 7 of the OSI model). The network load balancer works at layers 3 & 4 (network and transport layers). The network load balancer just forward requests whereas the application load balancer examines the contents of the HTTP request header to determine where to route the request.

When to use which Load balancer

- ALB supports path-based routing. You can configure rules for your listener that forward requests based on the URL in the request. This enables you to structure your application as smaller services, and route requests to the correct service based on the content of the URL. Primarily for http/https requests. -NLB has ability to handle volatile workloads and scale to millions of requests per second.

Configuring and using ALB

- Navigate to EC2 dashboard

- Launch 2 ec2 instances , they can be in different AZs . Install httpd server on top of it with different home page content

Step- 1

- Select load balancers from the left menu
- Click on create load balancer
- Select application load balancer
- Provide a name to the lb. And select scheme as internet facing
    - internet facing load balancer will be assigned a public ip hence accessible from internet
    - internal lb will be assigned a private ip
- Keep the default value of listener as it is . i.e. 80
    - listener decides on which port the lb will listen to requests
- in the availability zone tab , select the azs in which our instances are residing.

Step- 2

- Configure security settings . As of now we will keep it blank as we are not using tls incorparating acm.

Step- 3

- Choose a security group using which network could be opened to access elb

Step- 4

- Configure routing
- Here we create target group , which defines on which port the elb will route the requests to which servers (These can be created seperately and later be used here)
- Keep the default "create new target group selected" and give it a name
- Keep the target type as instance
- We'll keep the port as 80 as our httpd service runs on 80

Step- 4

- Register targets
- search the instances , click the check box and click on add to registered .
- If the instances are not added to registered , the requests wont be routed to the servers
- Click on review and create .
- ELB takes some time to create . Once setup , copy the dns or the a record from the description and test it in the browser. refresh to check if the requests are being routed to both servers

Things to check if the elb is not working

- httpd servie must be running on both servers
- ELB should listen to and forward requests on port 80
- Targets should be registered to target group

Best practices

- Use elb as way to expose certain application to internet . Instances can stay private and elb could be public
- Delete the elb after practice