AWS Load Balancer

What is a Load Balancer?

An AWS Load Balancer is a service that distributes incoming network traffic across multiple targets, such as EC2 instances, containers, or IP addresses. This helps ensure that no single server gets overwhelmed, improving the availability and reliability of your applications.

Types of AWS Load Balancers:

1. **Application Load Balancer (ALB)**:
   - Best for HTTP and HTTPS traffic.
   - Operates at the application layer (Layer 7).
   - Supports advanced routing, such as path-based and host-based routing.

2. **Network Load Balancer (NLB)**:
   - Best for TCP, UDP, and TLS traffic.
   - Operates at the transport layer (Layer 4).
   - Handles millions of requests per second with ultra-low latency.

3. **Gateway Load Balancer (GWLB)**:
   - Best for third-party virtual appliances like firewalls and intrusion detection systems.
   - Operates at the network layer.
   - Simplifies deployment, scaling, and management of virtual appliances.

4. **Classic Load Balancer (CLB)**:
   - Legacy load balancer that supports both HTTP/HTTPS and TCP protocols.
   - Operates at both the application layer and the transport layer.
   - Recommended to use ALB or NLB for new applications.

Why Use a Load Balancer?

1. **High Availability**: Distributes traffic across multiple targets to prevent overload and ensure availability.

2. **Scalability**: Automatically adjusts to handle varying traffic loads.

3. **Fault Tolerance**: Routes traffic away from unhealthy instances, improving reliability.

4. **Security**: Provides SSL termination and integrates with AWS security features.

How a Load Balancer Works:

1. **Incoming Traffic**: Traffic from clients (users) hits the load balancer.

2. **Traffic Distribution**: The load balancer distributes the traffic to multiple backend targets based on routing rules.

3. **Health Checks**: Continuously monitors the health of targets and routes traffic only to healthy ones.

4. **Responses**: Backend targets process the requests and send responses back to the clients through the load balancer.

Example Scenario:

Imagine you have a website with heavy traffic:

1. **Set Up an ALB**: Create an Application Load Balancer.

2. **Configure Listeners**: Set up listeners for HTTP (port 80) and HTTPS (port 443).

3. **Define Target Groups**: Group your EC2 instances, containers, or IP addresses.

4. **Create Routing Rules**: Direct traffic based on URL paths or hostnames.

5. **Deploy and Monitor**: Launch your application and monitor traffic distribution and instance health.

Visualizing:

Think of a load balancer as a smart traffic cop at a busy intersection:

- **Incoming Cars (Traffic)**: Vehicles coming from different directions.

- **Traffic Cop (Load Balancer)**: Directs cars to different lanes (servers) to prevent jams.

- **Lanes (Targets)**: Multiple roads (servers) that cars can take to reach their destination.

Benefits of Load Balancers:

1. **Improved Performance**: Balances load to optimize resource use and reduce response times.

2. **Enhanced Security**: Supports SSL/TLS termination and integrates with AWS security groups and WAF.

3. **Flexible Routing**: Advanced routing capabilities based on application needs.

Summary:

An AWS Load Balancer helps manage and distribute incoming traffic across multiple targets, ensuring your application is highly available, scalable, and fault-tolerant. Different types of load balancers cater to different traffic types and requirements.