AWS Elastic Load Balancer (ELB)

Introduction

AWS Elastic Load Balancer (ELB) is a **managed load balancing service** provided by Amazon Web Services that automatically distributes incoming application traffic across multiple targets such as EC2 instances, containers, IP addresses, and Lambda functions. ELB improves **availability, scalability, and fault tolerance** of applications running in AWS.

In real-world production environments, ELB is a critical component for building **highly available and reliable architectures**.

Why Do We Need a Load Balancer?

Without a load balancer:

- A single server failure can bring down the entire application
- Traffic spikes may cause performance issues
- No automatic failover

With ELB:

- Traffic is evenly distributed
- Failed instances are automatically removed
- Applications scale seamlessly

Types of Elastic Load Balancers

1. Application Load Balancer (ALB)

- Works at **Layer 7 (HTTP/HTTPS)**
- Supports **path-based and host-based routing**
- Commonly used for **web applications and microservices**
- Supports containers (ECS/EKS)

Example:

- /login → Auth service

- /products → Product service

---

2. Network Load Balancer (NLB)

- Works at **Layer 4 (TCP/UDP)**
- Extremely **high performance and low latency**
- Can handle millions of requests per second
- Used for **real-time and high-throughput applications**

---

3. Gateway Load Balancer (GWLB)

- Used for deploying and managing **virtual appliances**
- Works with firewalls, intrusion detection systems
- Operates at **Layer 3 (IP)**

---

4. Classic Load Balancer (CLB)

- Legacy load balancer
- Supports both Layer 4 and Layer 7
- **Not recommended** for new applications

Key Components of ELB

1. Listener

A listener checks for incoming connection requests using a specific **protocol and port** (for example, HTTP on port 80 or HTTPS on port 443).

---

2. Target Group

A target group contains registered targets such as:

- EC2 instances

- IP addresses
- Lambda functions

Traffic is routed to targets based on rules defined in the listener.

---

3. Health Checks

ELB continuously monitors the health of registered targets.

- Healthy targets receive traffic
- Unhealthy targets are automatically removed

This ensures **high availability**.

ELB Features

- Automatic traffic distribution
- Built-in health checks
- High availability across multiple Availability Zones
- SSL/TLS termination
- Integration with Auto Scaling
- Secure with Security Groups

ELB with Auto Scaling (Real-Time Scenario)

1. User traffic increases
2. ELB distributes traffic evenly
3. Auto Scaling launches new EC2 instances
4. ELB automatically registers new instances
5. When traffic decreases, instances are terminated

This provides **cost optimization and performance stability**.

Security Best Practices for ELB

- Use **HTTPS (SSL/TLS certificates)**
- Restrict inbound traffic using **Security Groups**
- Enable **access logs** for auditing
- Place ELB in **public subnet** and backend servers in **private subnet**

Common Use Cases

- Hosting highly available web applications
- Microservices architecture
- Handling traffic spikes during sales or events
- Blue-Green and Rolling deployments

ELB vs Auto Scaling (Quick Clarification)

- **ELB** distributes traffic
- **Auto Scaling** adjusts the number of instances

They work best **together**.

Advantages of ELB

- Fully managed service
- No manual intervention required
- Highly scalable
- Fault tolerant
- Cost-effective

Conclusion

AWS Elastic Load Balancer is a **core AWS service** for designing modern, scalable, and highly available cloud architectures. Whether you are hosting a simple website or a large enterprise application, ELB plays a vital role in ensuring **performance, reliability, and resilience**.

LinkedIn Caption (Optional)

🚀 **AWS Elastic Load Balancer (ELB) – Explained**

Today I documented one of the most important AWS services used in real-world production environments.

✅ Types of ELB
✅ How ELB works
✅ Real-time use cases
✅ Best practices

Learning cloud architecture step by step ☁️

#AWS #CloudComputing #DevOps #ELB #LoadBalancer #AWSLearning