🔐 SSL/TLS on AWS Elastic Load Balancer (ELB) – Explained

In **AWS**, **SSL/TLS** is used with **Elastic Load Balancers (ELB)** to secure client-to-server communication over **HTTPS (port 443)** or **TLS (for TCP)**. ELBs handle **SSL termination**, which means they decrypt SSL/TLS traffic before passing it to backend targets.

🔑 Key Concepts

🔒 What is SSL/TLS in AWS ELB?

- **TLS (Transport Layer Security)** is the modern protocol used to secure traffic (successor to SSL).
- ELBs use **SSL/TLS certificates** to enable **HTTPS** or **TLS-secured** connections.
- Certificates are managed using **AWS Certificate Manager (ACM)** or **IAM Certificate Store**.

🔧 How SSL/TLS Works in ELB (Simplified)

Client (browser) ---> HTTPS/TLS ---> [ELB with SSL cert] ---> HTTP/TCP ---> Backend servers

- Client connects securely to ELB (HTTPS).
- ELB **terminates the SSL connection**, decrypts the traffic.
- ELB forwards traffic to targets in **plain HTTP or TCP**, or **re-encrypts** if configured.

✅ Which Load Balancers Support SSL/TLS?

Load Balancer Type	Supports SSL/TLS	Notes
Application Load Balancer (ALB)	✅ Yes	Use HTTPS listener (Layer 7)
Network Load Balancer (NLB)	✅ Yes	Use TLS listener (Layer 4)
Classic Load Balancer (CLB)	✅ Yes	Use HTTPS listener (Layer 4/7)
Gateway Load Balancer (GWLB)	❌ No	No SSL/TLS support

📦 Using SSL/TLS on ELB: Step-by-Step

1. Get an SSL/TLS Certificate

- Use **AWS Certificate Manager (ACM)** to:
  - Request a free public certificate (for domains you own)
  - Or import your own certificate (third-party)

2. Create or Modify Your Load Balancer

- For **ALB**:
  - Create a **HTTPS listener (port 443)**
  - Choose SSL certificate from ACM
- For **NLB**:
  - Create a **TLS listener**
  - Choose a certificate from ACM or IAM

3. Configure Listener Rules

- ALB allows **content-based routing** (e.g., URL, hostname).
- Redirect HTTP (port 80) to HTTPS (port 443) for secure-only traffic.

4. Set SSL Policy (optional)

- Choose a **TLS version** and allowed cipher suites.
- Use recommended policies like:
  - ELBSecurityPolicy-TLS-1-2-Ext-2021-06
  - ELBSecurityPolicy-TLS-1-3-2021-06 (for ALB)

🧠 Common Scenarios

Use Case	Recommended LB + Config
Secure web traffic (HTTPS)	ALB with HTTPS listener + ACM cert
Secure TCP traffic (e.g., mail, custom)	NLB with TLS listener + ACM cert
Host multiple domains on one LB	ALB with **SNI** and multiple certificates
Encrypt all traffic end-to-end	TLS on ELB + TLS on backend servers
Use Let's Encrypt certificate	Import into ACM (not automated by AWS)

📌 Notes and Tips

- **ACM Certificates are free** and automatically renewed.
- Certificates must be in the **same AWS region** as the ELB.

- **SNI (Server Name Indication)** allows multiple domains (e.g., api.example.com, www.example.com) on a single listener.

- For **custom TLS policies**, always prefer TLS 1.2 or TLS 1.3 — older versions (1.0, 1.1) are deprecated.

LAB

#Configure your App Servers
#Configure the Target-Groups
#Create and attach the Load-Balancer with listener’s
#Now, access the app on the un-secured link

#Now, to convert this into a secured layer

- Go to Load-Balancer
- Go to Listeners
- Add Listeners
- Protocol - Select - HTTPS
- Forward Traffic - Target-Group
- SSL/TLS Cert - Create from ACM
- Attach the Cert

#Once the cert is updated, our protocol changes from http to https via SSL