Elastic Load Balancing

- Distributes incoming application or network traffic across multiple targets, such as EC2 instances, containers (ECS), Lambda functions, and IP addresses, in multiple Availability Zones.
- When you create a load balancer, you must specify one public subnet from at least two Availability Zones. You can specify only one public subnet per Availability Zone.
- **Features:**
  - Accepts incoming traffic from clients and routes requests to its registered targets.
  - Monitors the health of its registered targets and routes traffic only to healthy targets.
  - Enable deletion protection to prevent your load balancer from being deleted accidentally. Disabled by default.
  - Deleting ELB won't delete the instances registered to it.
  - Cross Zone Load Balancing – when enabled, each load balancer node distributes traffic across the registered targets in all enabled AZs.
  - Supports SSL Offloading which is a feature that allows the ELB to bypass the SSL termination by removing the SSL-based encryption from the incoming traffic.
- can privately access Elastic Load Balancing APIs from your Amazon Virtual Private Cloud (VPC) by creating VPC endpoints. With VPC endpoints, the routing between the VPC and Elastic Load Balancing APIs is handled by the AWS network without the need for an Internet gateway, network address translation (NAT) gateway, or virtual private network (VPN) connection.
- Four types of ELB – ALB, NLB, Gateway LB, Classic LB
- You cannot convert one load balancer type into another.

Application Load Balancer

- operates at the request level (layer 7)
- routes traffic to EC2 instances, containers, IP addresses, and Lambda functions based on the content of the request.
- Ideal for advanced load balancing of HTTP and HTTPS traffic.
- improves the security of application, by ensuring that the latest SSL/TLS ciphers and protocols are used at all times.
- At least 2 subnets must be specified when creating this type of load balancer.
- Cross-zone load balancing is already enabled by default in Application Load Balancer.
-
- **Features:**
  - **Mutual TLS Support :**
    - a protocol for two-way authentication between clients and servers that use x509 certificate based identities.
    - ALB authenticates client certs issued by 3rd party or private ACM.
    - ALB also proxy client certificate info to targets, which can be used for authorization.
  - **Automatic target weights (ATW)**
  - When using VPC, can create and manage security groups for additional security.
  - Can configure ALB as internet facing or non-internet facing
  - ALB supports Outposts
  - ALB offer SSL certificates through ACM and IAM.
  - Supports HTTP/2 and gRPC. Also IPv6
  - **TLS offloading :** traffic encryption between ALB and clients that initiate TLS/SSL sessions.
  - **Sticky sessions :** mechanism to route requests from same client to same target. ALB supports both duration-based cookies and application-based cookies.
  - **Request Tracing :** ALB injects custom identifier "X-Amzn-Trace-Id" HTTP header on all requests coming in to LB. This uniqueId can be used to uncover any performance or timing issues in application stack at granularity.
  - **Redirects:** ALB can redirect request from one URL to another URL. 3 types of redirects are supported.
    - http to http http://hostA to http://hostB
    - https to https : http://hostA to https://hostB, https://hostA:portA/pathA to https://hostB:portB/pathB
    - http to https : https://hostA to https://hostB

- **Fixed Response:** ALB can control which client requests to be sent to backend app. HTTP error response code or error message can be sent from LB itself without forwarding to backend.
- **Server Name Indication (SNI) :** extension to the TLS protocol, a client indicates the hostname to connect to at the start of the TLS handshake.
- **IP addresses as Targets :** can load balance any app hosted on AWS or on-premises using IP addresses of backend apps as targets.
- **Lambda functions as targets :** can register lambda function as a target to serve HTTP(s) requests, enabling users to access serverless application. You can build an entire website using Lambda functions or combine EC2 instances, containers, on-premises servers and Lambda functions to build applications.
- **Content based routing :** if an application is composed of several individual services, ALB can route a request based on content of the request such as host, Path URL, HTTP header, HTTP method, Query String or source IP.
  - Host based routing
  - Path-based routing
  - HTTP header based routing
  - HTTP method based routing
  - Query string parameter-based routing
  - Source IP address CIDR-based routing
- **Containerized Application Support :** ALB integrated with ECS provides fully managed container offering. Supports load balancing across multiple ports on a single Amazon EC2 instance.
- **AWS WAF** can be used to protect web apps on ALB.
- ALB supports **round-robin load-balancing algorithm.** Slow start mode -very useful for applications that depend on cache and need a warm-up period before being able to respond to requests with optimal performance.
- **User Authentication :** can offload authentication from apps into ALB. ALB can integrate with Amazon Cognito, to authenticate through social identity providers google, Facebook and amazon, through enterprise identity providers such Microsoft active directory via SAML or any OpenID connect.

- **Monitoring ALB :**
  - **CloudWatch Metric :** to retrieve statistics about data points for LB and targets as an ordered time-series data, known as metrics. Can use these metrics to verify if system is performing as expected.
  - **Access logs :** use this to capture detailed info about requests made to LB & store in as log file S3. Can use this log file to analyze traffic patterns and to troubleshoot issues with targets.
  - **Connection logs :** use connection logs to capture attributes about the requests sent to your load balancer, and store them as log files in Amazon S3. You can use these connection logs to determine the client IP address and port, client certificate information, connection results, and TLS ciphers being used. These connection logs can then be used to review request patterns, and other trends.
  - **Request tracing :** use this to tract http requests.LB adds a header with trace ID to every request it receives.
  - **CloudTrail logs :** use AWS CloudTrail to capture detailed information about the calls made to the Elastic Load Balancing API and store them as log files in Amazon S3, which can be used to determine which calls were made, the source IP address where the call came from, who made the call, when the call was made, and so on.
- **Troubleshooting ALB :**
  - If a target is taking longer than expected to enter the InService state, it might be failing health checks.
  - target is not in service until it passes one health check.
  - Check for below issues :
    - A security group does not allow traffic from LB using health check port and protocol.
    - A NACL doesn’t allow inbound traffic on the health check port and outbound traffic on the ephemeral ports (1024-65535).
    - Ping path doesn’t exist
    - Connection times out
  - If the load balancer is not responding to requests, check for the following issues:
    - internet-facing load balancer is attached to a private subnet
    - A security group or network ACL does not allow traffic

Network Load Balancer

- operates at the connection level (layer 4)
- routes traffic to EC2 instances, microservices, and containers within Amazon VPC, based on IP protocol data.
- Ideal for load balancing of TCP and UDP traffic. (TCP port- 1 to 1-65535)
- capable of handling **millions** of requests per second while maintaining ultra-low latencies.
- optimized to handle sudden and volatile traffic patterns while using a single static IP address per Availability Zone.
- NLB with TCP and TLS Listeners can be used to setup AWS PrivateLink. cannot set up PrivateLink with UDP .
- NLB idle timeout for TCP connections is 350 seconds. The idle timeout for UDP flows is 120 seconds.
- can enable cross-zone loading balancing only after creating your Network Load Balancer.
- Network Load Balancer currently supports 200 targets per Availability Zone
- **Features**
  - load balance TCP and UDP traffic, routing connections to targets – **EC2 instances, microservices, and containers.**
  - **TLS offloading :** NLB supports TLS client session termination. This enables to offload TLS termination task to LB, preserving source-Ip of backend app.
  - **Sticky sessions**
  - **Low latency :** NLB offers extremely low latencies for latency-sensitive apps.
  - **Preserve source IP address :** NLB can preserve client IP allowing backend-apps to see IP of client. Can be used for further processing by backend applications.
  - **Static IP support :** NLB automatically provides static IP per AZ.
  - **Elastic IP support :** NLB allows the option to assign elastic IP per AZ
  - **DNS fail-over :** If there are no healthy targets in a given AZ, then Route 53 routes traffic to nodes in another AZ.
  - **Integration with Route-53 :** If NLB is unresponsive then integration with Route53 will remove unavailable NLB IP address from service and direct traffic to an alternate NLB in another region.
  - **Long-lived TCP connections :** NLB supports long-lived connections, ideal for WebSocket type of apps.
  - **Zonal Isolation**

Gateway Load Balancer

- helps to easily deploy, scale, and manage third-party virtual appliances.
- one gateway for distributing traffic across multiple virtual appliances while scaling them, based on demand.
- can find, test, and buy virtual appliances from third-party vendors directly in AWS Marketplace.
- Gateway Load Balancer runs within one AZ.
- provides both Layer 3 gateway and Layer 4 load balancing capabilities.
- It is architected to handle millions of requests/second, volatile traffic patterns, introduces extremely low latency.
- By default, Gateway Load Balancer defines a flow as a combination of a 5-tuple that comprises Source IP, Destination IP, IP Protocol, Source Port, and Destination Port. Using the default 5-tuple hash, Gateway Load Balancer makes sure that both directions of a flow (i.e., source to destination, and destination to source) are consistently forwarded to the same target.
- Gateway Load Balancer idle timeout for TCP connections is 350 seconds. The idle timeout for non-TCP flows is 120 seconds. These timeouts are fixed and cannot be changed.
-
- **Features**
  - Scale your virtual appliance instances automatically
  - Bring higher availability to your third-party virtual appliances
  - Monitor continuous health and performance metrics
  - Simplify deployment with AWS Marketplace
  - Ensure private connectivity over the AWS network using Gateway Load Balancer Endpoints.

- To use static IP or PrivateLink on Application Load Balancer, forward traffic from NLB to ALB.
- To use a single ALB for handling HTTP and HTTPS requests, add listener for HTTP (80) and HTTPS (443).
- back-end server authentication is not supported with an ALB and NLB, only encryption is supported.

- To load balance applications distributed across a VPC and on-premises location, There are various ways to achieve hybrid load balancing. If an application runs on targets distributed between a VPC and an on-premises location, you can add them to the same target group using their IP addresses. To migrate to AWS without impacting your application, gradually add VPC targets to the target group and remove on-premises targets from the target group.
  If you have two different applications such that the targets for one application are in a VPC and the targets for other applications are in on-premises location, you can put the VPC targets in one target group and the on-premises targets in another target group and use content based routing to route traffic to each target group. You can also use separate load balancers for VPC and on-premises targets and use DNS weighting to achieve weighted load balancing between VPC and on-premises targets.
- Cannot have Network Load Balancer with a mix of ELB-provided IPs and Elastic IPs or assigned private Ips.
- Cannot assign more than one EIP to my Network Load Balancer in each subnet
- For each associated subnet that a load balancer is in, the Network Load Balancer can only support a single private IP.
- Can use combination of ALB, NLB, Classic LB as part of free tier. ALB NLB each for 15 LCUs, classic for 15GB resp.750 hours are shared across 3 LBs.