



AWS Elastic Load Balancer(ELB)

Khalid Bin Sattar



Agenda

- Overview of Amazon ELB.
 - Introduction to Amazon ELB.
 - How Load Balancer work?
 - Concept of Elastic Load Balancer.
 - Features of Elastic Load Balancing.
 - Types of Elastic Load Balancer.
 - Listeners.
 - Targets.
 - Limits.
 - Demo of Elastic Load Balancers.
- 



Overview of Amazon ELB

What Is Amazon ELB?

- Elastic Load Balancing automatically distributes your incoming application traffic across multiple targets, such as EC2 instances. It monitors the health of registered targets and routes traffic only to the healthy targets. Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers, and Classic Load Balancers.
- The load balancer serves as a single point of contact for clients, which increases the availability of your application. You can add and remove instances from your load balancer as your needs change, without disrupting the overall flow of requests to your application. Elastic Load Balancing scales your load balancer as traffic to your application changes over time, and can scale to the vast majority of workloads automatically.



Overview of Amazon ELB

What Is Amazon ELB?

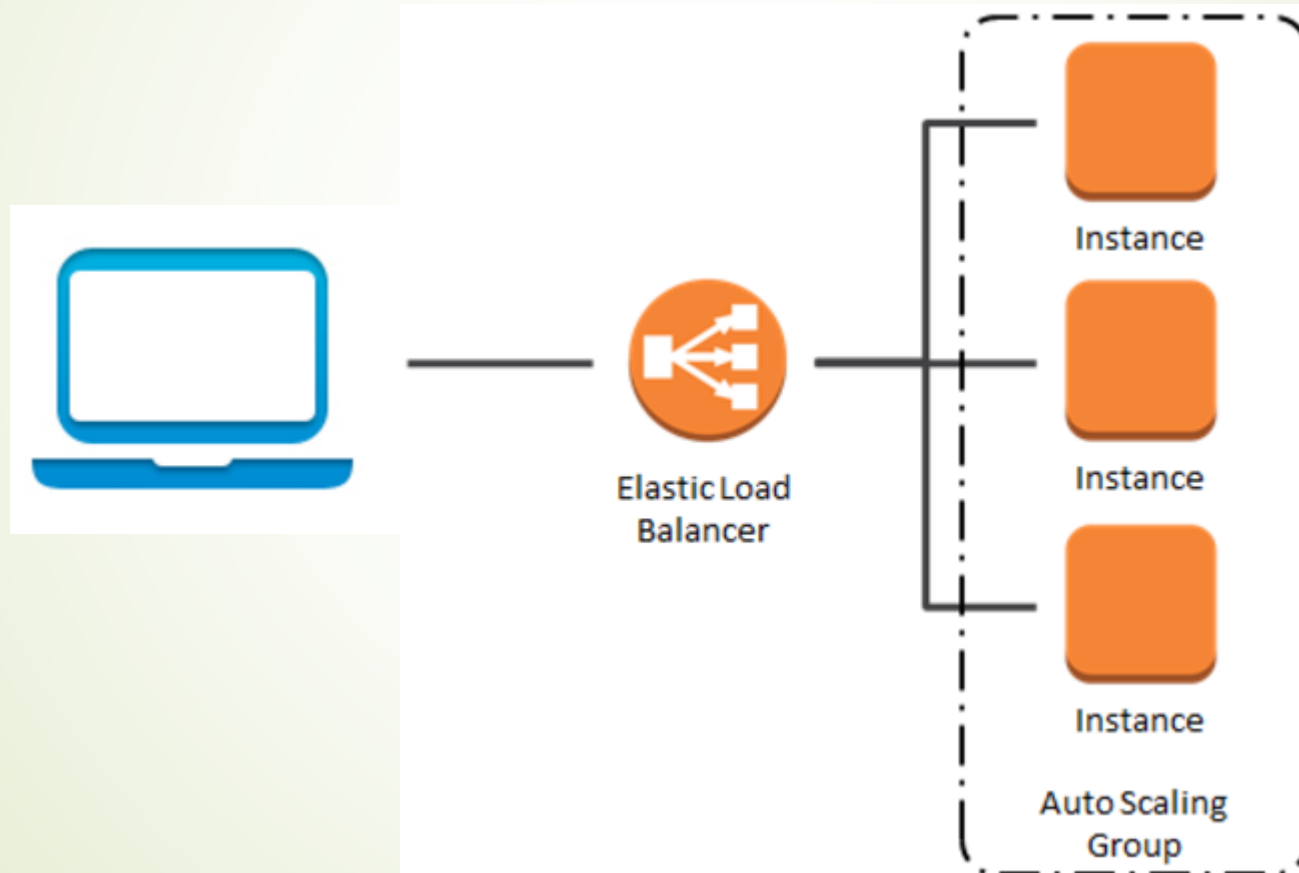
- You can configure health checks, which are used to monitor the health of the registered instances so that the load balancer can send requests only to the healthy instances. You can also offload the work of encryption and decryption to your load balancer so that your instances can focus on their main work.
- Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers, and Classic Load Balancers. You can select a load balancer based on your application needs.
- This increases the fault tolerance of your applications.



Architecture of Elastic Load Balancer(ELB)

Overview of Amazon ELB

Architecture of Amazon ELB.



Features of Elastic Load Balancer(ELB)





Introduction to Amazon ELB

Features of Elastic Load Balancing:-

1. High Availability

- Elastic Load Balancing automatically distributes traffic across multiple targets – Amazon EC2 instances, containers and IP addresses – in a single Availability Zone or multiple Availability Zones.

2. Health Checks

- Elastic Load Balancing can detect unhealthy targets, stop sending traffic to them, and then spread the load across the remaining healthy targets.

3. Security Features

- Use Amazon Virtual Private Cloud (Amazon VPC) to create and manage security groups associated with load balancers to provide additional networking and security options. You can also create an internal (non-internet-facing) load balancer.

4. Operational Monitoring

- Elastic Load Balancing provides integration with Amazon CloudWatch metrics and request tracing in order to monitor performance of your applications in real time.

How Elastic Load Balancing Works



Introduction to Amazon ELB

How Elastic Load Balancing Works:-

- A load balancer accepts incoming traffic from clients and routes requests to its registered targets (such as EC2 instances) in one or more Availability Zones. The load balancer also monitors the health of its registered targets and ensures that it routes traffic only to healthy targets. When the load balancer detects an unhealthy target, it stops routing traffic to that target, and then resumes routing traffic to that target when it detects that the target is healthy again.
- You configure your load balancer to accept incoming traffic by specifying one or more listeners. A listener is a process that checks for connection requests. It is configured with a protocol and port number for connections from clients to the load balancer and a protocol and port number for connections from the load balancer to the targets.
- Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers, and Classic Load Balancers. There is a key difference between the way you configure these load balancers. With Application Load Balancers and Network Load Balancers, you register targets in target groups, and route traffic to the target groups. With Classic Load Balancers, you register instances with the load balancer.



Concepts of Amazon ELB



Concepts of Amazon ELB

Availability Zones and Load Balancer Nodes:-

- When you enable an Availability Zone for your load balancer, Elastic Load Balancing creates a load balancer node in the Availability Zone. If you register targets in an Availability Zone but do not enable the Availability Zone, these registered targets do not receive traffic. Note that your load balancer is most effective if you ensure that each enabled Availability Zone has at least one registered target.
- AWS recommend that you enable multiple Availability Zones. With this configuration, if one Availability Zone becomes unavailable or has no healthy targets, the load balancer can continue to route traffic to the healthy targets in another Availability Zone.
- After you disable an Availability Zone, the targets in that Availability Zone remain registered with the load balancer, but the load balancer will not route traffic to them.



Concepts of Amazon ELB

Cross-Zone Load Balancing:-

- If the nodes for your load balancer can distribute requests regardless of Availability Zone, this is known as cross-zone load balancing. With cross-zone load balancing, the load balancer distributes traffic evenly across all registered targets in all enabled Availability Zones. Otherwise, each load balancer node distributes traffic only to registered targets in its Availability Zone.
- For example, suppose that you have 10 instances in us-west-2a and 2 instances in us-west-2b. With cross-zone load balancing, the load balancer distributes incoming requests evenly across all 12 instances. Otherwise, the 2 instances in us-west-2b serve the same amount of traffic as the 10 instances in us-west-2a.
- With Application Load Balancers, cross-zone load balancing is always enabled.
- With Network Load Balancers, each load balancer node distributes traffic across the registered targets in its Availability Zone only.



Concepts of Amazon ELB

Request Routing:-

- Before a client sends a request to your load balancer, it resolves the load balancer's domain name using a Domain Name System (DNS) server. The DNS entry is controlled by Amazon, because your load balancers are in the amazonaws.com domain. The Amazon DNS servers return one or more IP addresses to the client, which are the IP addresses of the load balancer nodes for your load balancer. As traffic to your application changes over time, Elastic Load Balancing scales your load balancer and updates the DNS entry. Note that the DNS entry also specifies the time-to-live (TTL) as 60 seconds, which ensures that the IP addresses can be remapped quickly in response to changing traffic.
- The client determines which IP address to use to send requests to the load balancer. The load balancer node that receives the request selects a healthy registered targets and sends the request to the target using its private IP address.



Concepts of Amazon ELB

Routing Algorithm:-

- With Application Load Balancers, the load balancer node that receives the request evaluates the listener rules in priority order to determine which rule to apply, and then selects a target from the target group for the rule action using the round robin routing algorithm. Routing is performed independently for each target group, even when a target is registered with multiple target groups.
- With Network Load Balancers, the load balancer node that receives the connection selects a target from the target group for the default rule using a flow hash routing algorithm.
- With Classic Load Balancers, the load balancer node that receives the request selects a registered instance using the round robin routing algorithm for TCP listeners and the least outstanding requests routing algorithm for HTTP and HTTPS listeners.



Concepts of Load Balancer Scheme



Concepts of Amazon ELB

Load Balancer Scheme:-

- When you create a load balancer, you must choose whether to make it an internal load balancer or an Internet-facing load balancer. Note that when you create a Classic Load Balancer in EC2-Classic, it must be an Internet-facing load balancer.
- The nodes of an Internet-facing load balancer have public IP addresses. The DNS name of an Internet-facing load balancer is publicly resolvable to the public IP addresses of the nodes. Therefore, Internet-facing load balancers can route requests from clients over the Internet.
- The nodes of an internal load balancer have only private IP addresses. The DNS name of an internal load balancer is publicly resolvable to the private IP addresses of the nodes. Therefore, internal load balancers can only route requests from clients with access to the VPC for the load balancer.



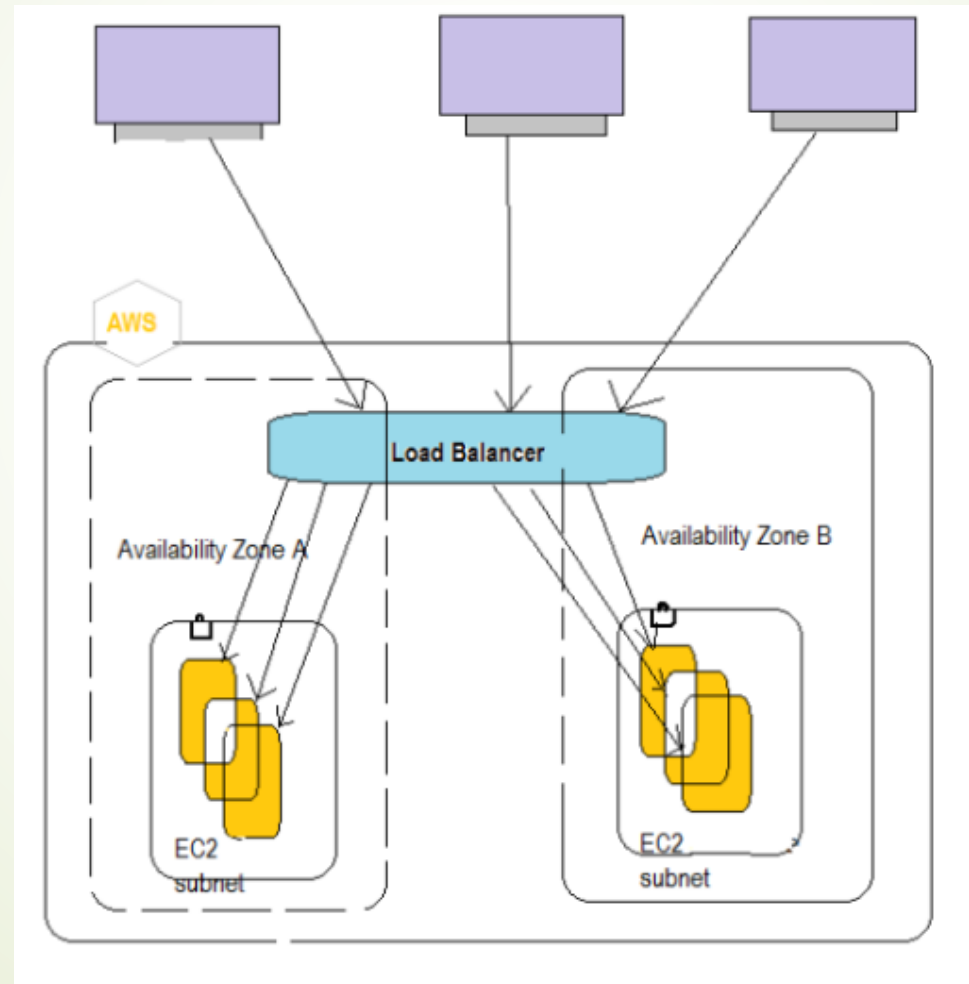
Concepts of Amazon ELB

Load Balancer Scheme:-

- Note that both Internet-facing and internal load balancers route requests to your targets using private IP addresses. Therefore, your targets do not need public IP addresses to receive requests from an internal or an Internet-facing load balancer.
- If your application has multiple tiers, for example web servers that must be connected to the Internet and database servers that are only connected to the web servers, you can design an architecture that uses both internal and Internet-facing load balancers. Create an Internet-facing load balancer and register the web servers with it. Create an internal load balancer and register the database servers with it. The web servers receive requests from the Internet-facing load balancer and send requests for the database servers to the internal load balancer. The database servers receive requests from the internal load balancer.

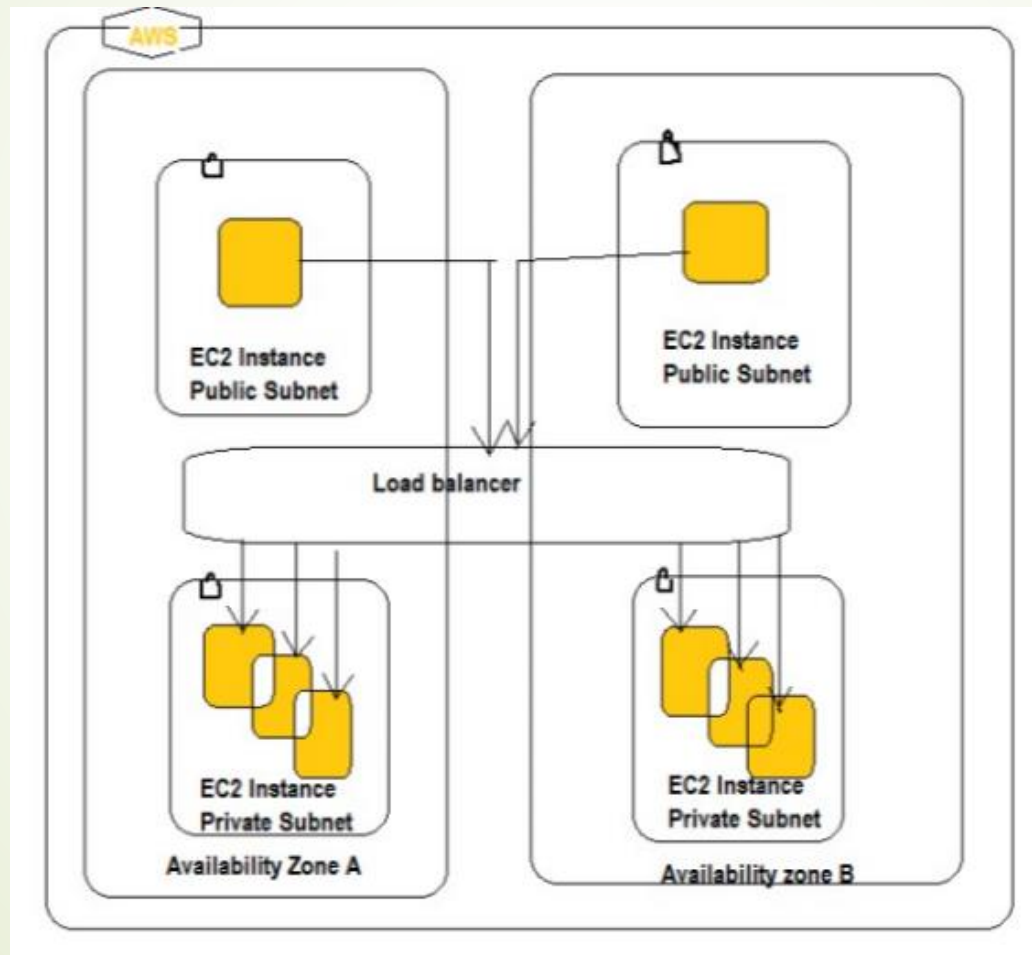
Concepts of Amazon ELB

Architecture of Internet Facing Load Balancer:-



Concepts of Amazon ELB

Architecture of Internal Facing Load Balancer:-

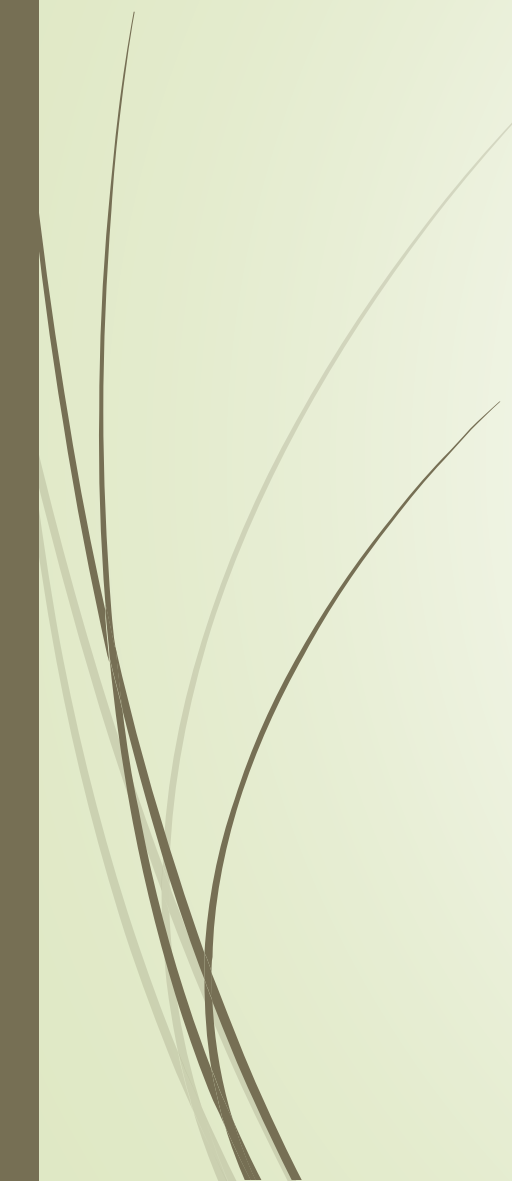




Application Load Balancer Overview

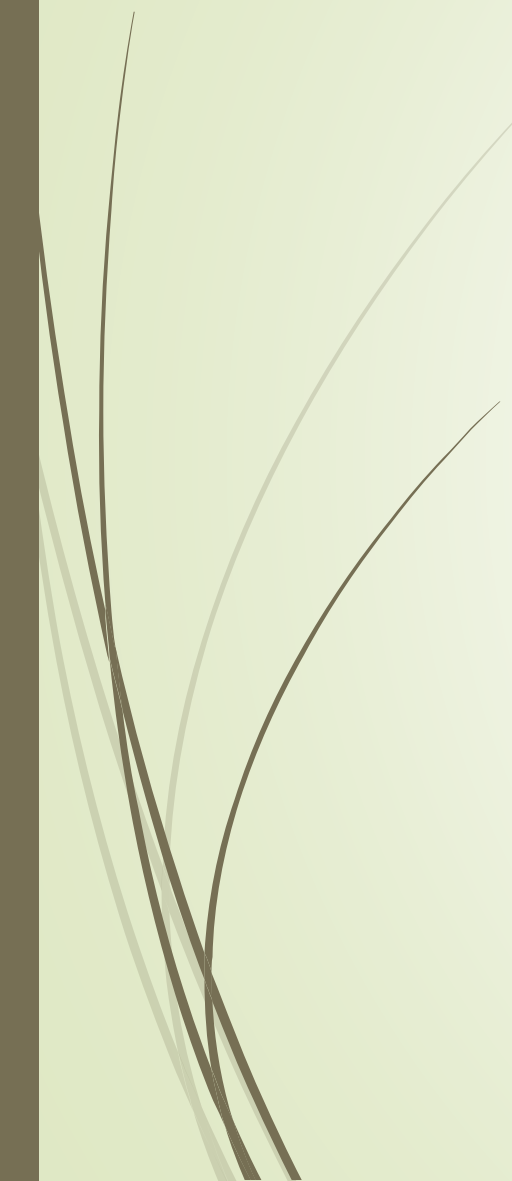


Application Load Balancer Overview

- 
- An Application Load Balancer functions at the application layer, the seventh layer of the Open Systems Interconnection (OSI) model. After the load balancer receives a request, it evaluates the listener rules in priority order to determine which rule to apply, and then selects a target from the target group for the rule action using the round robin routing algorithm. Note that you can configure listener rules to route requests to different target groups based on the content of the application traffic. Routing is performed independently for each target group, even when a target is registered with multiple target groups.
 - The load balancer distributes incoming application traffic across multiple targets, such as EC2 instances, in multiple Availability Zones. This increases the fault tolerance of your applications. Elastic Load Balancing detects unhealthy targets and routes traffic only to healthy targets.



Application Load Balancer Overview

- 
- The load balancer serves as a single point of contact for clients. This increases the availability of your application. You can add and remove targets from your load balancer as your needs change, without disrupting the overall flow of requests to your application. Elastic Load Balancing scales your load balancer as traffic to your application changes over time. Elastic Load Balancing can scale to the vast majority of workloads automatically.
 - You can configure health checks, which are used to monitor the health of the registered targets so that the load balancer can send requests only to the healthy targets.

Components of Application Load Balancer





Application Load Balancer Overview

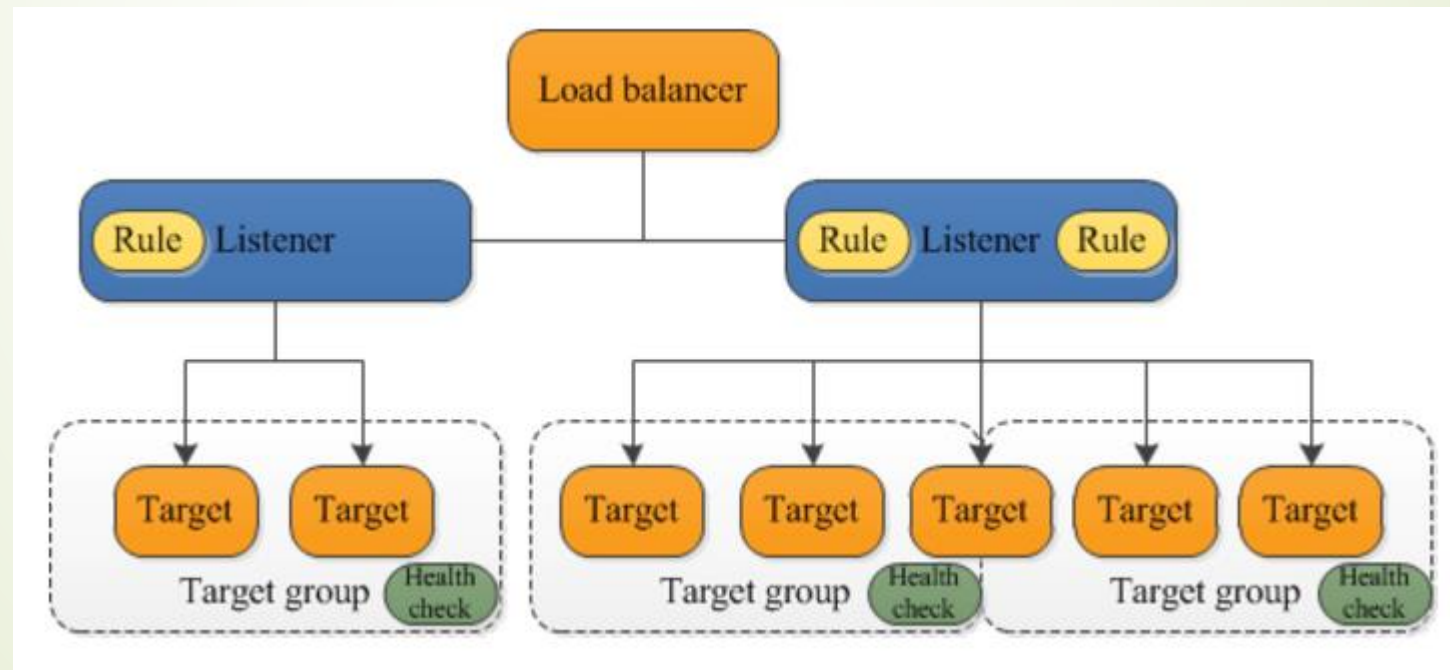
Components:-

- The load balancer serves as the single point of contact for clients. You add one or more listeners to your load balancer.
- A listener checks for connection requests from clients, using the protocol and port that you configure, and forwards requests to one or more target groups, based on the rules that you define. Each rule specifies a target group, condition, and priority. When the condition is met, the traffic is forwarded to the target group. You must define a default rule for each listener, and you can add rules that specify different target groups based on the content of the request (also known as content-based routing).
- Each target group routes requests to one or more registered targets, such as EC2 instances, using the protocol and port number that you specify. You can register a target with multiple target groups. You can configure health checks on a per target group basis. Health checks are performed on all targets registered to a target group that is specified in a listener rule for your load balancer.

Application Load Balancer Overview

Components:-

The following diagram illustrates the basic components. Notice that each listener contains a default rule, and one listener contains another rule that routes requests to a different target group. One target is registered with two target groups.



Benefits of Application Load Balancer





Application Load Balancer Overview

Benefits of Application Load Balancer:-

- Support for 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.
- Support for routing requests to multiple services on a single EC2 instance by registering the instance using multiple ports.
- Support for containerized applications. Amazon EC2 Container Service (Amazon ECS) can select an unused port when scheduling a task and register the task with a target group using this port. This enables you to make efficient use of your clusters.



Application Load Balancer Overview

Benefits of Application Load Balancer:-

- Support for monitoring the health of each service independently, as health checks are defined at the target group level and many CloudWatch metrics are reported at the target group level. Attaching a target group to an Auto Scaling group enables you to scale each service dynamically based on demand.
- Access logs contain additional information and are stored in compressed format.
- Improved load balancer performance.

Getting Started with Application Load Balancers






Application Load Balancer Overview

This tutorial provides a hands-on introduction to Application Load Balancers through the AWS Management Console, a web-based interface. To create your first Application Load Balancer, complete the following steps.

Topics:-

- Step 1: Select a Load Balancer Type.
 - Step 2: Configure Your Load Balancer and Listener.
 - Step 3: Configure a Security Group for Your Load Balancer.
 - Step 4: Configure Your Target Group.
 - Step 5: Register Targets with Your Target Group.
 - Step 6: Create and Test Your Load Balancer.
 - Step 7: Delete Your Load Balancer (Optional).
- 

Description of Application Load Balancers





Description of Application Load Balancers

A load balancer serves as the single point of contact for clients. Clients send requests to the load balancer, and the load balancer sends them to targets, such as EC2 instances, in two or more Availability Zones. To configure your load balancer, you create target groups, and then register targets with your target groups. You also create listeners to check for connection requests from clients, and listener rules to route requests from clients to the targets in one or more target groups.

Contents:-

- Load Balancer Security Groups.
- Load Balancer State.
- Load Balancer Attributes.
- IP Address Type.
- Deletion Protection.
- Connection Idle Timeout.
- Create a Load Balancer.
- Update Availability Zones.
- Update Security Groups.
- Update the Address Type.
- Update Tags.
- Delete a Load Balancer.



Description of Application Load Balancers

Load Balancer Security Groups :-

- A security group acts as a firewall that controls the traffic allowed to and from your load balancer. You can choose the ports and protocols to allow for both inbound and outbound traffic.
- The rules for the security groups associated with your load balancer security group must allow traffic in both directions on both the listener and the health check ports. Whenever you add a listener to a load balancer or update the health check port for a target group, you must review your security group rules to ensure that they allow traffic on the new port in both directions.



Description of Application Load Balancers

Load Balancer State:-

A load balancer can be in one of the following states:

1. provisioning

- The load balancer is being set up.

2. active

- The load balancer is fully set up and ready to route traffic.

3. failed

- The load balancer could not be set up.

Description of Application Load Balancers

Load Balancer Attributes:-

The following are the load balancer attributes:

1. `access_logs.s3.enabled`
 - Indicates whether access logs stored in Amazon S3 are enabled.
2. `access_logs.s3.bucket`
 - The name of the S3 bucket for the access logs. For more information, see [Bucket Permissions](#).
3. `access_logs.s3.prefix`
 - The prefix for the location in the S3 bucket. If you don't specify a prefix, the access logs are stored in the root of the bucket.
4. `deletion_protection.enabled`
 - Indicates whether deletion protection is enabled.
5. `idle_timeout.timeout_seconds`
 - The idle timeout value, in seconds. The default is 60 seconds.

Description of Application Load Balancers

IP Address Type:-

You can set the IP address type of your Internet-facing load balancer when you create it or after it is active. Note that internal load balancers must use IPv4 addresses.

The following are the load balancer IP address types:

- Ipv4:-The load balancer supports only IPv4 addresses (for example, 192.0.2.1)
- Dualstack:-The load balancer supports both IPv4 and IPv6 addresses (for example, 2001:0db8:85a3:0:0:8a2e:0370:7334).

Clients that communicate with the load balancer using IPv4 addresses resolve the A record and clients that communicate with the load balancer using IPv6 addresses resolve the AAAA record. However, the load balancer communicates with its targets using IPv4 addresses, regardless of how the client communicates with the load balancer.



Description of Application Load Balancers

Deletion Protection:-

- To prevent your load balancer from being deleted accidentally, you can enable deletion protection. By default, deletion protection is disabled for your load balancer.
- If you enable deletion protection for your load balancer, you must disable it before you can delete the load balancer.

Description of Application Load Balancers

Connection Idle Timeout:-

- For each request that a client makes through a load balancer, the load balancer maintains two connections. A front-end connection is between a client and the load balancer, and a back-end connection is between the load balancer and a target. For each front-end connection, the load balancer manages an idle timeout that is triggered when no data is sent over the connection for a specified time period. If no data has been sent or received by the time that the idle timeout period elapses, the load balancer closes the front-end connection.
- By default, Elastic Load Balancing sets the idle timeout value to 60 seconds. Therefore, if the target doesn't send some data at least every 60 seconds while the request is in flight, the load balancer can close the front-end connection. To ensure that lengthy operations such as file uploads have time to complete, send at least 1 byte of data before each idle timeout period elapses, and increase the length of the idle timeout period as needed.
- For back-end connections, AWS recommend that you enable the keep-alive option for your EC2 instances. You can enable keep-alive in your web server settings or in the kernel settings for your EC2 instances. Keep-alive, when enabled, enables the instances to tear down back-end connections when an operation is finished.



Create an Application Load Balancers



Create an Application Load Balancers

A load balancer takes requests from clients and distributes them across targets in a target group.

Ensure that you have a virtual private cloud (VPC) with at least one public subnet in each of the Availability Zones used by your targets.

Below are the Steps to Create and update the Properties of Load Balancer:-

- Step 1: Create a Load Balancer.
- Step 2: Update Availability Zone.
- Step 3: Update Security Groups.
- Step 4: Update Address Types.
- Step 5: Update Tags.
- Step 6: Delete a Load Balancer.



Listeners for Your Application Load Balancers



Listeners for Your Application Load Balancers

Before you start using your Application Load Balancer, you must add one or more listeners. A listener is a process that checks for connection requests, using the protocol and port that you configure. The rules that you define for a listener determine how the load balancer routes requests to the targets in one or more target groups.

Contents:-

- Listener Configuration.
- Listener Rules.
- Host Conditions.
- Path Conditions.



Listeners for Your Application Load Balancers

Listener Configuration:-

Listeners support the following protocols and ports:

- Protocols: HTTP, HTTPS
- Ports: 1-65535

You can use an HTTPS listener to offload the work of encryption and decryption to your load balancer so that your targets can focus on their main work. If the listener protocol is HTTPS, you must deploy exactly one SSL server certificate on the listener.

Application Load Balancers provide native support for Websockets. You can use WebSockets with both HTTP and HTTPS listeners.



Listeners for Your Application Load Balancers

Listener Rules:-

Each listener has a default rule, and you can optionally define additional rules. Each rule consists of a priority, action, optional host condition, and optional path condition.

- **Default Rules:-**When you create a listener, you define an action for the default rule. Default rules can't have conditions. If no conditions for any of a listener's rules are met, then the action for the default rule is taken.
- **Rule Priority:-**Each rule has a priority. Rules are evaluated in priority order, from the lowest value to the highest value. The default rule has lowest priority. You can change the priorities of the nondefault rules at any time.
- **Rule Actions:-**Each rule action has a type and a target group. Currently, the only supported type is forward, which forwards requests to the target group. You can change the target group for a rule at any time.

Listeners for Your Application Load Balancers

Rule Conditions:-

There are two types of rule conditions: host and path. When the conditions for a rule are met, then its action is taken.

1. Host Conditions:-

- You can use host conditions to define rules that forward requests to different target groups based on the host name in the host header (also known as host-based routing). This enables you to support multiple domains using a single load balancer.
- Each host condition has one hostname. If the hostname in the host header matches the hostname in a listener rule exactly, the request is routed using that rule.
- A hostname is case-insensitive, can be up to 128 characters in length, and can contain any of the following characters. Note that you can include up to three wildcard characters.
- Example hostnames:- example.com, test.example.com and *.example.com

Listeners for Your Application Load Balancers

Rule Conditions:-

2. Path Conditions:-

- You can use path conditions to define rules that forward requests to different target groups based on the URL in the request (also known as path-based routing).
- Each path condition has one path pattern. If the URL in a request matches the path pattern in a listener rule exactly, the request is routed using that rule.
- A path pattern is case-sensitive, can be up to 128 characters in length, and can contain any of the following characters. Note that you can include up to three wildcard characters.
- Example path patterns :- `/img/*` and `/js/*`

Target Groups for Your Application Load Balancers



Target Group for Your Application Load Balancers

You register targets with a target group. To route requests to the targets in a target group, specify the target group in a rule for one of the listeners for your load balancer.

You define health check settings for your load balancer on a per target group basis. Each target group uses the default health check settings, unless you override them when you create the target group or modify them later on. After you specify a target group in a rule for a listener, the load balancer continually monitors the health of all targets registered with the target group that are in an Availability Zone enabled for the load balancer. The load balancer routes requests to the registered targets that are healthy.

Contents:-

- Routing Configuration.
- Target Type.
- Registered Targets.
- Target Group Attributes.
- Deregistration Delay.
- Sticky Sessions.
- Create a Target Group.
- Health Checks for Your Target Groups.
- Register Targets with Your Target Group.
- Delete a Target Group.



Target Group for Your Application Load Balancers

Routing Configuration:-

- By default, a load balancer routes requests to its targets using the protocol and port number that you specified when you created the target group. Alternatively, you can override the port used for routing traffic to a target when you register it with the target group.

Target groups support the following protocols and ports:-

- Protocols: HTTP, HTTPS
- Ports: 1-65535
- If a target group is configured with the HTTPS protocol or uses HTTPS health checks, SSL connections to the targets use the security settings from the ELBSecurityPolicy2016-08 policy.



Target Group for Your Application Load Balancers

Target Type:-

When you create a target group, you specify its target type, which determines how you specify its targets. After you create a target group, you cannot change its target type.

The following are the possible target types:

- Instance:- The targets are specified by instance ID.
- Ip:- The targets are specified by IP address.



Target Group for Your Application Load Balancers

Registered Targets:-

- Your load balancer serves as a single point of contact for clients and distributes incoming traffic across its healthy registered targets. You can register each target with one or more target groups. You can register each EC2 instance or IP address with a single target group multiple times using different ports, which enables the load balancer to route requests to ECS containers.
- If demand on your application increases, you can register additional targets with one or more target groups in order to handle the demand. The load balancer starts routing requests to a newly registered target as soon as the registration process completes and the target passes the initial health checks.
- If demand on your application decreases, or you need to service your targets, you can deregister targets from your target groups. Deregistering a target removes it from your target group, but does not affect the target otherwise. The load balancer stops routing requests to a target as soon as it is deregistered. The target enters the draining state until in-flight requests have completed. You can register the target with the target group again when you are ready for it to resume receiving requests.

Target Group for Your Application Load Balancers

Target Group Attributes:-

The following are the target group attributes:

- `deregistration_delay.timeout_seconds`:- The amount of time for Elastic Load Balancing to wait before deregistering a target. The range is 0-3600 seconds. The default value is 300 seconds.
- `stickiness.enabled`:- Indicates whether sticky sessions are enabled.
- `stickiness.lb_cookie.duration_seconds`:- The cookie expiration period, in seconds. After this period, the cookie is considered stale. The minimum value is 1 second and the maximum value is 7 days (604800 seconds). The default value is 1 day (86400 seconds).
- `stickiness.type`:- The type of stickiness. The possible value is `lb_cookie`.



Target Group for Your Application Load Balancers

Deregistration Delay:-

- Elastic Load Balancing stops sending requests to targets that are deregistering. By default, Elastic Load Balancing waits 300 seconds before completing the deregistration process, which can help in-flight requests to the target to complete. To change the amount of time that Elastic Load Balancing waits, update the deregistration delay value. Note that you can specify a value of up to 1 hour, and that Elastic Load Balancing waits the full amount of time specified, regardless of whether there are in-flight requests.
- If a deregistering target terminates the connection before the deregistration delay elapses, the client receives a 500-level error response.
- The initial state of a deregistering target is draining. After the deregistration delay elapses, the deregistration process completes and the state of the target is unused. If the target is part of an Auto Scaling group, it can be terminated and replaced. However, connections between load balancer nodes and a deregistering target are kept for up to one hour if there are in-flight requests.

Limits for Your Application Load Balancers





Limits for Your Application Load Balancers

1. Regional Limits:-

- Load balancers per region: 20.
- Target groups per region: 3000.

2. Load Balancer Limits:-

- Listeners per load balancer: 50
- Targets per load balancer: 1000
- Subnets per Availability Zone per load balancer: 1
- Security groups per load balancer: 5
- Rules per load balancer (not counting default rules): 100
- Number of times a target can be registered per load balancer: 100



Limits for Your Application Load Balancers

3. Listener Limits:-

- Certificates per listener: 1

4. Target Group Limits:-

- Load balancers per target group: 1
- Targets per target group: 1000

5. Rule Limits:-

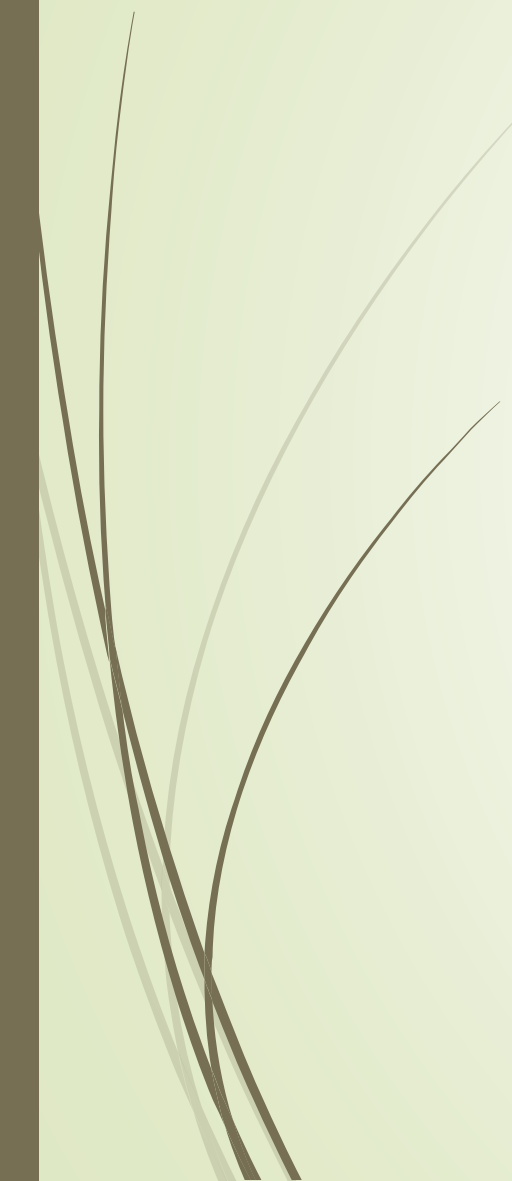
- Conditions per rule: 2 (one host condition, one path condition)
- Actions per rule: 1
- Target groups per action: 1



Network Load Balancer Overview




Network Load Balancer Overview

- 
- A Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. The load balancer distributes incoming traffic across multiple targets, such as EC2 instances. It can handle millions of requests per second. After the load balancer receives a connection, it selects a target from the target group for the default rule using a flow hash routing algorithm. It attempts to open a TCP connection to the selected target on the port specified in the listener configuration. It forwards the request without modifying the headers.
 - When you enable an Availability Zone for the load balancer, Elastic Load Balancing creates a load balancer node in the Availability Zone. Each load balancer node for your Network Load Balancer distributes traffic across the registered targets in its Availability Zone only. If you enable multiple Availability Zones for your load balancer, this increases the fault tolerance of your applications. If all of your targets in one Availability Zone are unhealthy and you have registered targets in other Availability Zones, your Network Load Balancer automatically routes traffic to the healthy targets in the other Availability Zones.



Network Load Balancer Overview

- 
- The load balancer serves as a single entry point for clients. This increases the availability of your application. You can add and remove targets from your load balancer as your needs change, without disrupting the overall flow of requests to your application. Elastic Load Balancing scales your load balancer as traffic to your application changes over time. Elastic Load Balancing can scale to the vast majority of workloads automatically.
 - You can configure health checks, which are used to monitor the health of the registered targets so that the load balancer can send requests only to the healthy targets.



Network Load Balancer Overview

Components:-

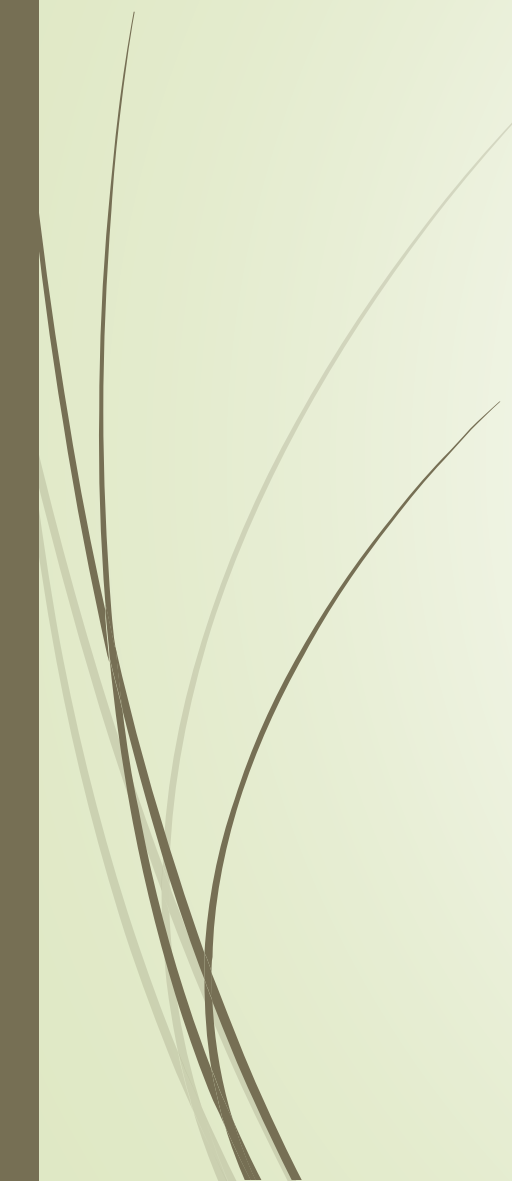
- The load balancer serves as the single point of contact for clients. You add one or more listeners to your load balancer.
- A listener checks for connection requests from clients, using the protocol and port that you configure, and forwards requests to a target group.
- Each target group routes requests to one or more registered targets, such as EC2 instances, using the TCP protocol and the port number that you specify. You can register a target with multiple target groups. You can configure health checks on a per target group basis. Health checks are performed on all targets registered to a target group that is specified in a listener rule for your load balancer.



Benefits of Network Load Balancer



Benefits of Network Load Balancer

- 
1. Ability to handle volatile workloads and scale to millions of requests per second.
 2. Support for fixed IP addresses for the load balancer. You can assign one Elastic IP address per subnet enabled for the load balancer.
 3. Source IP addresses are preserved and provided to your applications.
 4. Support for routing requests to multiple services on a single EC2 instance by registering the instance using multiple ports.
 5. Support for containerized applications. Amazon EC2 Container Service (Amazon ECS) can select an unused port when scheduling a task and register the task with a target group using this port. This enables you to make efficient use of your clusters.
 6. Support for monitoring the health of each service independently, as health checks are defined at the target group level and many CloudWatch metrics are reported at the target group level. Attaching a target group to an Auto Scaling group enables you to scale each service dynamically based on demand.

Getting Started with Network Load Balancers

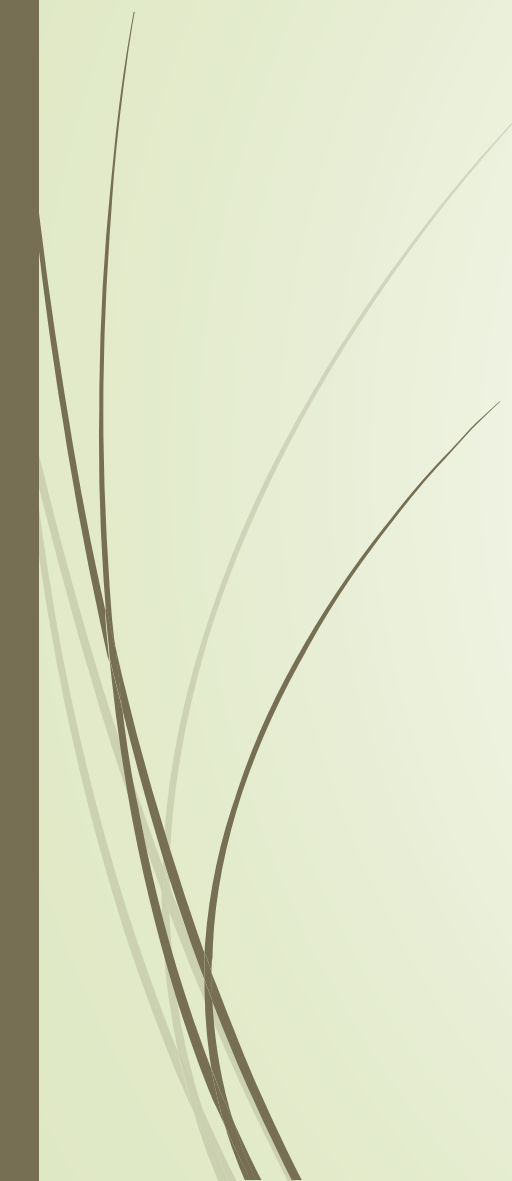




Getting Started with Network Load Balancers

This tutorial provides a hands-on introduction to Network Load Balancers through the AWS Management Console, a web-based interface. To create your first Network Load Balancer, complete the following steps.

Tasks:-

- Step 1: Select a Load Balancer Type.
 - Step 2: Configure Your Load Balancer and Listener.
 - Step 3: Configure Your Target Group.
 - Step 4: Register Targets with Your Target Group.
 - Step 5: Create and Test Your Load Balancer.
 - Step 6: Delete Your Load Balancer (Optional).
- 

Description of Network Load Balancers





Description of Network Load Balancers

Contents:-

- Load Balancer Security Groups.
- Load Balancer State.
- Load Balancer Attributes.
- IP Address Type.
- Deletion Protection.
- Connection Idle Timeout.
- Create a Load Balancer.
- Update Availability Zones.
- Update Security Groups.
- Update the Address Type.
- Update Tags.
- Delete a Load Balancer.

Limits for Your Network Load Balancers





Limits for Your Network Load Balancers

1. Regional Limits:-

- Network Load Balancers per region: 20
- Target groups per region: 3000

2. Load Balancer Limits

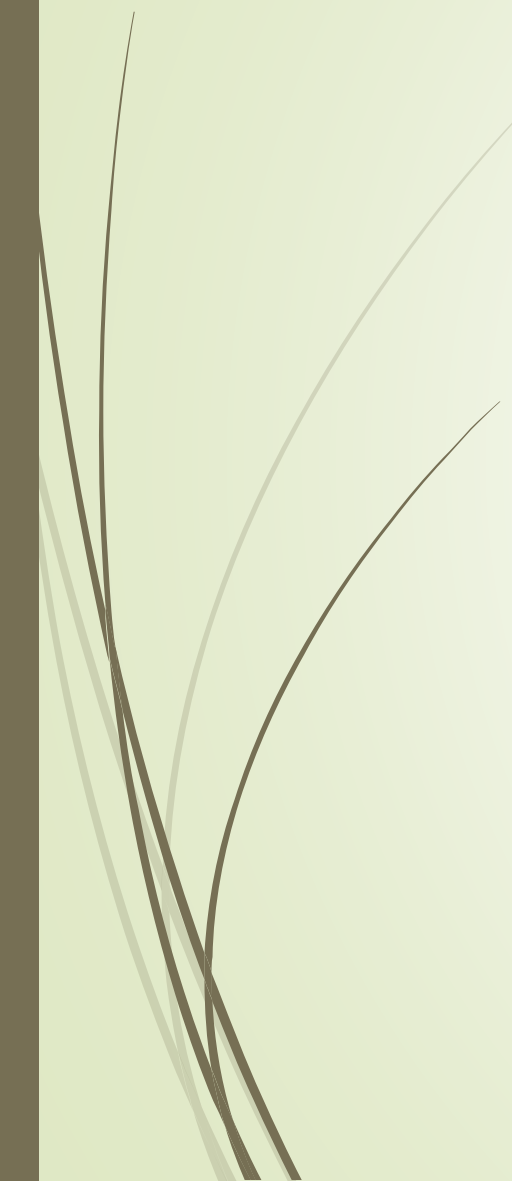
- Listeners per load balancer: 50
- Subnets per Availability Zone per load balancer: 1
- Targets per load balancer per Availability Zone: 200
- Load balancers per target group: 1



Classic Load Balancer Overview

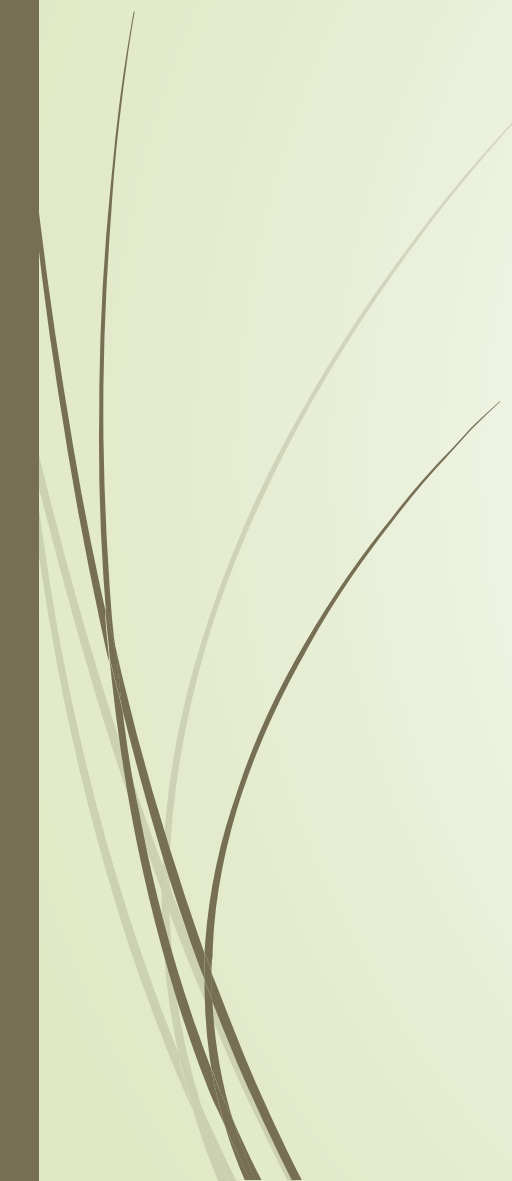


Classic Load Balancer Overview

- 
- A load balancer distributes incoming application traffic across multiple EC2 instances in multiple Availability Zones. This increases the fault tolerance of your applications. Elastic Load Balancing detects unhealthy instances and routes traffic only to healthy instances.
 - Your load balancer serves as a single point of contact for clients. This increases the availability of your application. You can add and remove instances from your load balancer as your needs change, without disrupting the overall flow of requests to your application. Elastic Load Balancing scales your load balancer as traffic to your application changes over time. Elastic Load Balancing can scale to the vast majority of workloads automatically.
 - A listener checks for connection requests from clients, using the protocol and port that you configure, and forwards requests to one or more registered instances using the protocol and port number that you configure. You add one or more listeners to your load balancer.



Classic Load Balancer Overview

- 
- You can configure health checks, which are used to monitor the health of the registered instances so that the load balancer can send requests only to the healthy instances.
 - To ensure that your registered instances are able to handle the request load in each Availability Zone, it is important to keep approximately the same number of instances in each Availability Zone registered with the load balancer. For example, if you have ten instances in Availability Zone us-west-2a and two instances in us-west-2b, the requests are distributed evenly between the two Availability Zones. As a result, the two instances in us-west-2b serve the same amount of traffic as the ten instances in us-west-2a. Instead, you should have six instances in each Availability Zone.
 - By default, the load balancer distributes traffic evenly across the Availability Zones that you enable for your load balancer. To distribute traffic evenly across all registered instances in all enabled Availability Zones, enable cross-zone load balancing on your load balancer. However, AWS recommends that you maintain approximately equivalent numbers of instances in each Availability Zone for better fault tolerance.



Benefits of Classic Load Balancer



Benefits of Classic Load Balancer

Using a Classic Load Balancer instead of an Application Load Balancer has the following benefits:

- Support for EC2-Classic
- Support for TCP and SSL listeners
- Support for sticky sessions using application-generated cookies.

Tutorial: Create a Classic Load Balancer





Benefits of Classic Load Balancer

This tutorial provides a hands-on introduction to Classic Load Balancers through the AWS Management Console, a web-based interface. You'll create a load balancer that receives public HTTP traffic and sends it to your EC2 instances.

Note that you can create your load balancer for use with EC2-Classical or a VPC. Some of the tasks described in this tutorial apply only to load balancers in a VPC.

- Step 1: Select a Load Balancer Type.
- Step 2: Define Your Load Balancer.
- Step 3: Assign Security Groups to Your Load Balancer in a VPC.
- Step 4: Configure Health Checks for Your EC2 Instances.
- Step 5: Register EC2 Instances with Your Load Balancer.
- Step 6: Tag Your Load Balancer (Optional).
- Step 7: Create and Verify Your Load Balancer.
- Step 8: Delete Your Load Balancer (Optional).

Public DNS Names for Your Load Balancer





Public DNS Names for Your Load Balancer

When your load balancer is created, it receives a public DNS name that clients can use to send requests. The DNS servers resolve the DNS name of your load balancer to the public IP addresses of the load balancer nodes for your load balancer. Each load balancer node is connected to the back-end instances using private IP addresses.

1. EC2-VPC:-Load balancers in a VPC support IPv4 addresses only. The console displays a public DNS name with the following form:

`name-1234567890.region.elb.amazonaws.com`

2. EC2-Classic:- Load balancers in EC2-Classic support both IPv4 and IPv6 addresses. The console displays the following public DNS names:

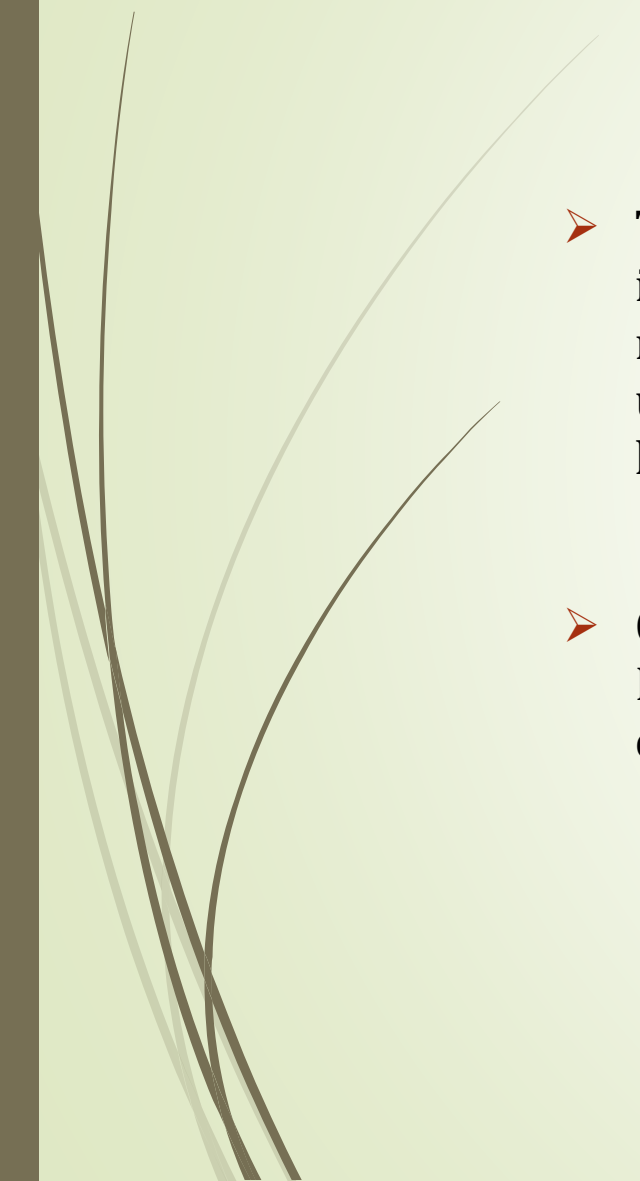
`name-123456789.region.elb.amazonaws.com`

`ipv6.name-123456789.region.elb.amazonaws.com`

`dualstack.name-123456789.region.elb.amazonaws.com`



Public DNS Names for Your Load Balancer

- 
- The base public DNS name returns only IPv4 records. The public DNS name with the ipv6 prefix returns only IPv6 records. The public DNS name with the dualstack prefix returns both IPv4 and IPv6 records. AWS recommend that you enable IPv6 support by using the DNS name with the dualstack prefix to ensure that clients can access the load balancer using either IPv4 or IPv6.
 - Clients can connect to your load balancer in EC2-Classic using either IPv4 or IPv6. However, communication between the load balancer and its back-end instances uses only IPv4, regardless of how the client communicates with your load balancer.

Registered Instances for Your Classic Load Balancer





Public DNS Names for Your Load Balancer

After you've created your Classic Load Balancer, you must register your EC2 instances with the load balancer. You can select EC2 instances from a single Availability Zone or multiple Availability Zones within the same region as the load balancer. Elastic Load Balancing routinely performs health checks on registered EC2 instances, and automatically distributes incoming requests to the DNS name of your load balancer across the registered, healthy EC2 instances.

Contents:-

- Prepare Your VPC and EC2 Instances.
- Configure Health Checks for Your Classic Load Balancer.
- Configure Security Groups for Your Classic Load Balancer.
- Add or Remove Availability Zones for Your Load Balancer in EC2-Classic.
- Add or Remove Subnets for Your Classic Load Balancer in a VPC.
- Register or Deregister EC2 Instances for Your Classic Load Balancer.



Limits for Your Classic Load Balancer



Limits for Your Classic Load Balancer

Your AWS account has the following limits related to Classic Load Balancers.

- Load balancers per region: 20
- Listeners per load balancer: 100
- Security groups per load balancer: 5
- Subnets per Availability Zone per load balancer: 1



AWS Elastic Load Balancer Pricings



AWS Elastic Load Balancer Pricings

Check the Video for the ELB Pricings

