**Load Balancing:**

Load balancing is the method of distributing network traffic equally across a pool of resources that support an application. Modern applications must process millions of users simultaneously and return the correct text, videos, images, and other data to each user in a fast and reliable manner. To handle such high volumes of traffic, most applications have many resource servers with duplicate data between them. A load balancer is a device that sits between the user and the server group and acts as an invisible facilitator, ensuring that all resource servers are used equally.

Benefits of load balancing:

Application availability:

Server failure or maintenance can increase application downtime, making your application unavailable to visitors. Load balancers increase the fault tolerance of your systems by automatically detecting server problems and redirecting client traffic to available servers.

• Run application server maintenance or upgrades without application downtime

• Provide automatic disaster recovery to backup sites

• Perform health checks and prevent issues that can cause downtime

Application scalability:

You can use load balancers to direct network traffic intelligently among multiple servers. Your applications can handle thousands of client requests because load balancing does the following:

• Prevents traffic bottlenecks at any one server

• Predicts application traffic so that you can add or remove different servers, if needed

• Adds redundancy to your system so that you can scale with confidence

Application security:

Load balancers come with built-in security features to add another layer of security to your internet applications. They are a useful tool to deal with distributed denial of service attacks, in which attackers flood an application server with millions of concurrent requests that cause server failure. Load balancers can also do the following:

* Monitor traffic and block malicious content
* Automatically redirect attack traffic to multiple backend servers to minimize impact
* Route traffic through a group of network firewalls for additional security

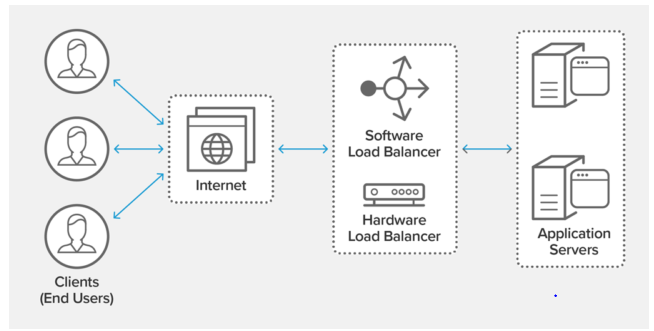
Application performance:

Load balancers improve application performance by increasing response time and reducing network latency. They perform several critical tasks such as the following:

* Distribute the load evenly between servers to improve application performance
* Redirect client requests to a geographically closer server to reduce latency
* Ensure the reliability and performance of physical and virtual computing resources

Application session Persistance:

This is the ability to make sure that a user’s session data goes to one server throughout the user’s session. If the server changes midway, it will cause performance issues and the data will not be saved. Being able to handle tons of data being saved is one huge benefit if you know how to



Types of load Balancers:

Application Load Balancer:

An Application Load Balancer makes routing decisions at the application layer (HTTP/HTTPS), supports path-based routing, and can route requests to one or more ports on each container instance in your cluster. Application Load Balancers support dynamic host port mapping.

Network Load Balancer:

A load balancer serves as the single point of contact for clients. The load balancer distributes incoming traffic across multiple targets, such as Amazon EC2 instances. This increases the availability of your application.

Classic Load Balancer:

Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level. Classic Load Balancer is intended for applications that are built within the EC2-Classic network.

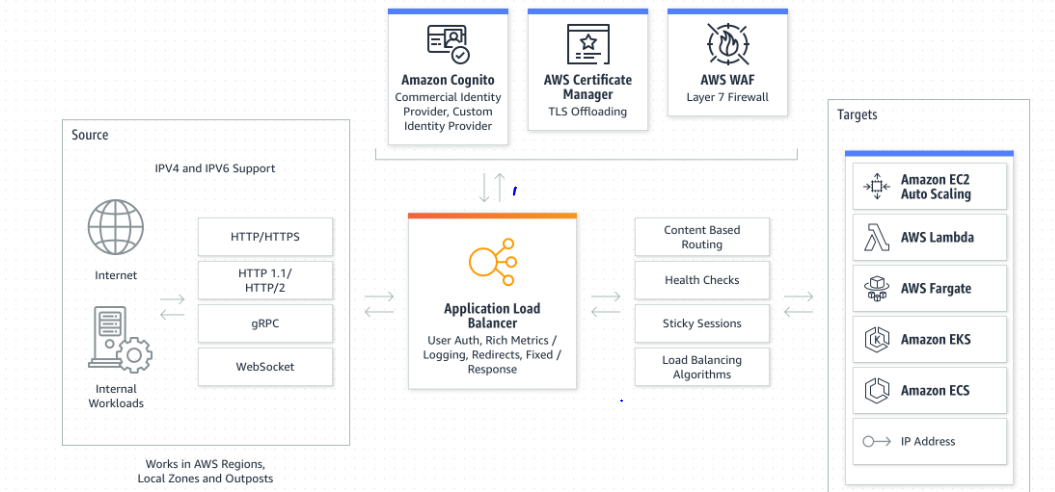
Load Balancing with High availability:

**Kubernetes load Balancing:**

A Kubernetes load balancer is a component that distributes network traffic across multiple instances of an application running in a K8S cluster. It acts as a traffic manager, ensuring that incoming requests are evenly distributed among the available instances to optimize performance and prevent overload on any single instance, providing high availability and scalability. Load balancers in K8S can be implemented by using a cloud provider-specific load balancer such as Azure Load Balancer, AWS Network Load Balancer (NLB), or Elastic Load Balancer (ELB) that operates at the Network Layer 4 of the OSI model. In addition, many different ingress controllers exist that can be installed into the K8S cluster. Each provides different features and can be configured to perform different load-balancing distribution like NGINX, HAProxy, Istio Ingress, and Traefik.

**Load Balancing using Elastic load balancer (AWS):**

Elastic Load Balancing automatically distributes your incoming traffic across multiple targets, such as EC2 instances, containers, and IP addresses, in one or more Availability Zones. It monitors the health of its registered targets and routes traffic only to the healthy targets. An Elastic Load Balancer is highly available. You can distribute incoming traffic across your Amazon EC2 instances in a single Availability Zone or multiple Availability Zones. An Elastic Load Balancer automatically scales its request handling capacity in response to incoming application traffic. To ensure that your targets are available and healthy, Elastic Load Balancer runs health checks on targets on a configurable cadence



**AWS ALB**