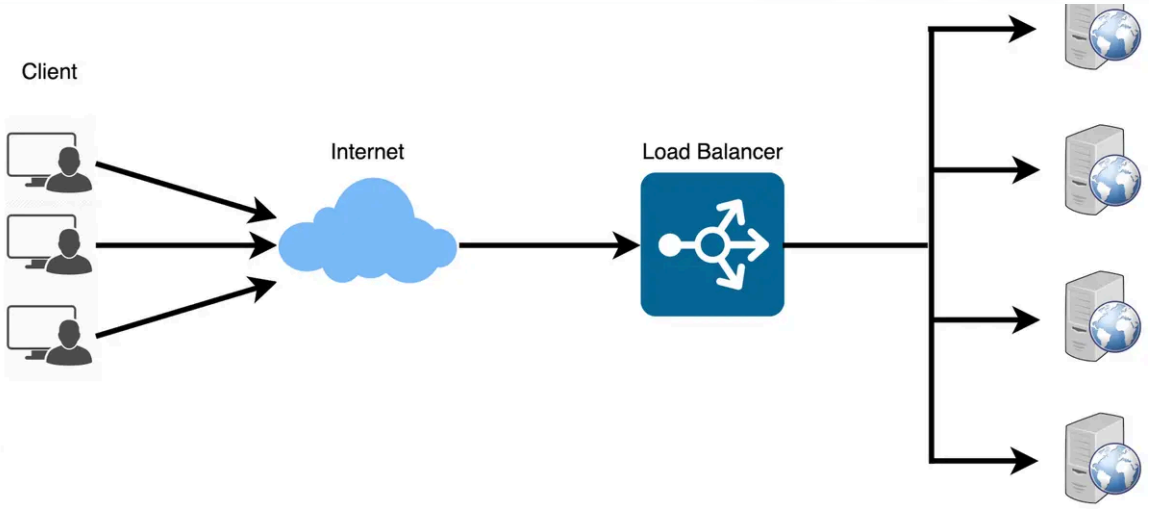
Introduction to Load Balancing

ON THIS PAGE

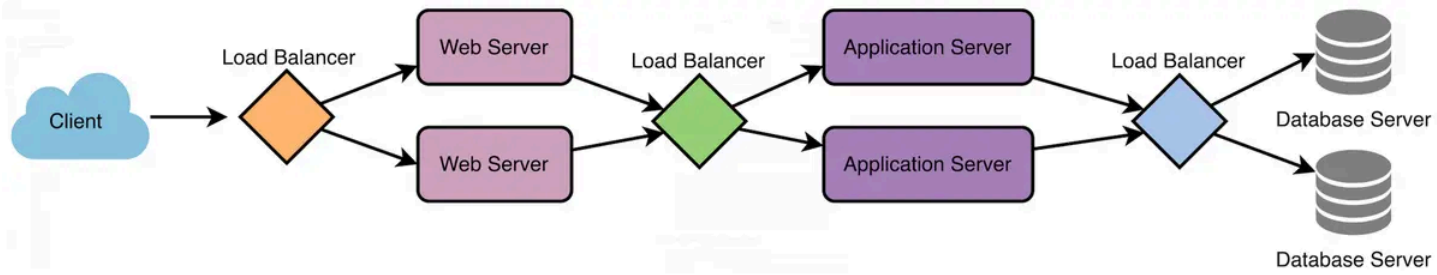
Load balancing is a crucial component of System Design, as it helps distribute incoming requests and traffic evenly across multiple servers. The main goal of load balancing is to ensure high availability, reliability, and performance by avoiding overloading a single server and avoiding downtime.

Typically a load balancer sits between the client and the server accepting incoming network and application traffic and distributing the traffic across multiple backend servers using various algorithms. By balancing application requests across multiple servers, a load balancer reduces the load on individual servers and prevents any one server from becoming a single point of failure, thus improving overall application availability and responsiveness.



To utilize full scalability and redundancy, we can try to balance the load at each layer of the system. We can add LBs at three places:

- Between the user and the web server
- Between web servers and an internal platform layer, like application servers or cache servers
- Between internal platform layer and database.



Key terminology and concepts

Load Balancer: A device or software that distributes network traffic across multiple servers based on predefined rules or algorithms.

Load Balancing Algorithm: The method used by the load balancer to determine how to distribute incoming traffic among the backend servers.

Health Checks: Periodic tests performed by the load balancer to determine the availability and performance of backend servers. Unhealthy servers are removed from the server pool until they recover.

Session Persistence: A technique used to ensure that subsequent requests from the same client are directed to the same backend server, maintaining session state and providing a consistent user experience.

SSL/TLS Termination: The process of decrypting SSL/TLS-encrypted traffic at the load balancer level, offloading the decryption burden from backend servers and allowing for centralized SSL/TLS management.

How Load Balancer works?

Load balancers work by distributing incoming network traffic across multiple servers or resources to ensure efficient utilization of computing resources and prevent overload. Here are the general steps that a load balancer follows to distribute traffic:

1. The load balancer receives a request from a client or user.
2. The load balancer evaluates the incoming request and determines which server or resource should handle the request. This is done based on a predefined load-balancing algorithm that takes into account factors such as server capacity, server response time, number of active connections, and geographic location.
3. The load balancer forwards the incoming traffic to the selected server or resource.
4. The server or resource processes the request and sends a response back to the load balancer.

5. The load balancer receives the response from the server or resource and sends it to the client or user who made the request.

← Previous

Introduction To System Design

Next →

Load Balancing Algorithms

Mark as Completed