What Is Load Balancing?

An Introduction to Load Balancing

Load Balancing Definition: Load balancing is the process of distributing network traffic across multiple servers. This ensures no single server bears too much demand. By spreading the work evenly, load balancing improves application responsiveness. It also increases availability of applications and websites for users. Modern applications cannot run without load balancers. Over time, software load balancers have added additional capabilities including security and application.

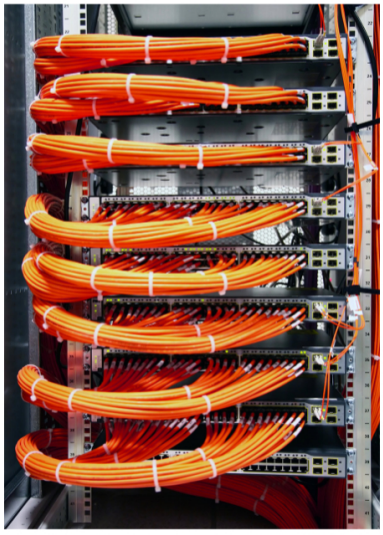
About Load Balancers

As an organization meets demand for its applications, the load balancer decides which servers can handle that traffic. This maintains a good user experience.

Load balancers manage the flow of information between the server and an endpoint device (PC, laptop, tablet or smartphone). The server could be on-premises, in a data center or the public cloud. The server can also be physical or virtualized. The load balancer helps servers move data efficiently, optimizes the use of application delivery resources and prevents server overloads. Load balancers conduct continuous health checks on servers to ensure they can handle requests. If necessary, the load balancer removes unhealthy servers from the pool until they are restored. Some load balancers even trigger the creation of new virtualized application servers to cope with increased demand.

Traditionally, load balancers consist of a hardware appliance. Yet they are increasingly becoming software-defined. This is why load balancers are an essential part of an organization's digital strategy.

History of Load Balancing



Load balancing got its start in the 1990s as hardware appliances distributing traffic across a network. Organizations wanted to improve accessibility of applications running on servers. Eventually, load balancing took on more responsibilities with the advent of Application Delivery Controllers (ADCs). They provide security along with seamless access to applications at peak times.

ADCs fall into three categories: hardware appliances, virtual appliances (essentially the software extracted from legacy hardware) and software-native load balancers. As computing moves to the cloud, software ADCs perform similar tasks to hardware. They also come with added functionality and flexibility. They let an organization quickly and securely scale up its application services based on demand in the cloud. Modern ADCs allow organizations to consolidate network-based services. Those services include SSL/TLS offload, caching, compression, intrusion detection and web application firewalls. This creates even shorter delivery times and greater scalability.

Load Balancing and SSL

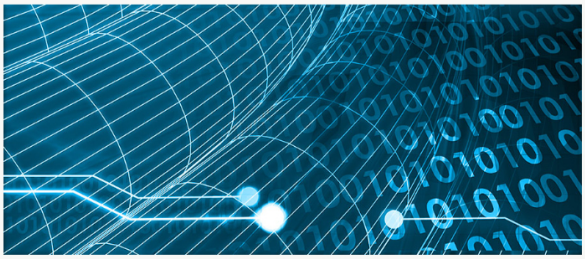
Secure Sockets Layer (SSL) is the standard security technology for establishing an encrypted link between a web server and a browser. SSL traffic is often decrypted at the load balancer. When a load balancer decrypts traffic before passing the request on, it is called SSL termination. The load balancer saves the web servers from having to expend the extra CPU cycles required for decryption. This improves application performance.

However, SSL termination comes with a security concern. The traffic between the load balancers and the web servers is no longer encrypted. This can expose the application to possible attack. However, the risk is lessened when the load balancer is within the same data center as the web servers.

Another solution is the SSL pass-through. The load balancer merely passes an encrypted request to the web server. Then the web server does the decryption. This uses more CPU power on the web server. But organizations that require extra security may find the extra overhead worthwhile.

Load Balancing and Security

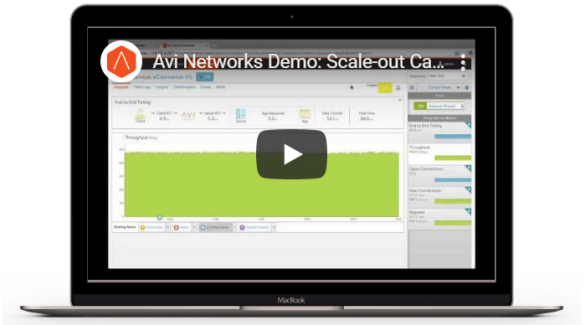
Load Balancing plays an important security role as computing moves evermore to the cloud. The off-loading function of a load balancer defends an organization against distributed denial-of-service (DDoS) attacks. It does this by shifting attack traffic from the corporate server to a public cloud provider. DDoS attacks represent a large portion of cybercrime as their number and size continues to rise. Hardware defense, such as a perimeter firewall, can be costly and require significant maintenance. Software load balancers with cloud offload provide efficient and cost-effective protection.



Load Balancing Algorithms

There is a variety of load balancing methods, which use different algorithms best suited for a particular situation.

- Least Connection Method — directs traffic to the server with the fewest active connections. Most useful when there are a large number of persistent connections in the traffic unevenly distributed between the servers.
- Least Response Time Method — directs traffic to the server with the fewest active connections and the lowest average response time



average response time.

- Round Robin Method — rotates servers by directing traffic to the first available server and then moves that server to the bottom of the queue. Most useful when servers are of equal specification and there are not many persistent connections.
- IP Hash — the IP address of the client determines which server receives the request.

Load balancing has become a necessity as applications become more complex, user demand grows and traffic volume increases. Load balancers allow organizations to build flexible networks that can meet new challenges without compromising security, service or performance.

Load Balancing Benefits

Load balancing can do more than just act as a network traffic cop. Software load balancers provide benefits like predictive analytics that determine traffic bottlenecks before they happen. As a result, the software load balancer gives an organization actionable insights. These are key to automation and can help drive business decisions.

In the seven-layer Open System Interconnection (OSI) model, network firewalls are at levels one to three (L1-Physical Wiring, L2-Data Link and L3-Network). Meanwhile, load balancing happens between layers four to seven (L4-Transport, L5-Session, L6-Presentation and L7-Application).

Load balancers have different capabilities, which include:

- L4 — directs traffic based on data from network and transport layer protocols, such as IP address and TCP port.
- L7 — adds content switching to load balancing. This allows routing decisions based on attributes like HTTP header, uniform resource identifier, SSL session ID and HTML form data.
- GSLB — Global Server Load Balancing extends L4 and L7 capabilities to servers in different geographic locations.

More enterprises are seeking to deploy cloud-native applications in data centers and public clouds. This is leading to significant changes in the capability of load balancers. In turn, this creates both challenges and opportunities for infrastructure and operations leaders.

Software load balancers provide predictive analytics that determine traffic bottlenecks before they happen.

Actionable insights by load balancers that can help drive business decisions.

Global Server Load Balancing extends L4 and L7 load balancing capabilities to servers in different geographic locations.

Software Load Balancers vs. Hardware Load Balancers

Load balancers run as hardware appliances or are software-defined. Hardware appliances often run proprietary software optimized to run on custom processors. As traffic increases, the vendor simply adds more load balancing appliances to handle the volume. Software defined load balancers usually run on less-expensive, standard Intel x86 hardware. Installing the software in cloud environments like AWS EC2 eliminates the need for a physical appliance.

SOFTWARE PROS

- Flexibility to adjust for changing needs.
- Ability to scale beyond initial capacity by adding more software instances.
- Lower cost than purchasing and maintaining physical machines. Software can run on any standard device, which tends to be cheaper.
- Allows for load balancing in the cloud, which provides a managed, off-site solution that can draw resources from an elastic network of servers. Cloud computing also allows for the flexibility of hybrid hosted and in-house solutions. The main load balancer could be in-house while the backup is a cloud load balancer.

SOFTWARE CONS

- When scaling beyond initial capacity, there can be some delay while configuring load balancer software.
- Ongoing costs for upgrades.

HARDWARE PROS

- Fast throughput due to software running on specialized processors.
- Increased security since only the organization can access the servers physically.
- Fixed cost once purchased.

HARDWARE CONS

- Require more staff and expertise to configure and program the physical machines.
- Inability to scale when the set limit on number of connections has been made. Connections are refused or service degraded until additional machines are purchased and installed.
- Higher cost for purchase and maintenance of physical network load balancer. Owning a hardware load balancer may also require paying for consultants to manage it.

Types of Load Balancing

- **SDN** — Load balancing using SDN (software-defined networking) separates the control plane from the data plane for application delivery. This allows the control of multiple load balancing. It also helps the network to function like the virtualized versions of compute and storage. With the centralized control, networking policies and parameters can be programmed directly for more responsive and efficient application services. This is how networks can become more agile.
- **UDP** — A UDP load balancer utilizes User Datagram Protocol (UDP). UDP load balancing is often used for live broadcasts and online games when speed is important and there is little need for error correction. UDP has low latency because it does not provide time-consuming health checks.
- **TCP** — A TCP load balancer uses transmission control protocol (TCP). TCP load balancing provides a reliable and error-checked stream of packets to IP addresses, which can otherwise easily be lost or corrupted.
- **SLB**— Server Load Balancing (SLB) provides network services and content delivery using a series of load balancing algorithms. It prioritizes responses to the specific requests from clients over the network. Server load balancing distributes client traffic to servers to ensure consistent, high-performance application delivery.
- **Virtual** — Virtual load balancing aims to mimic software-driven infrastructure through virtualization. It runs the software of a physical load balancing appliance on a virtual machine. Virtual load balancers, however, do not avoid the architectural challenges of traditional hardware appliances which include limited scalability and automation, and lack of central management.
- **Elastic** — Elastic Load Balancing scales traffic to an application as demand changes over time. It uses system health checks to learn the status of application pool members (application servers) and routes traffic appropriately to available servers, manages fail-over to high availability targets, or automatically spins-up

additional capacity.

- **Geographic** — Geographic load balancing redistributes application traffic across data centers in different locations for maximum efficiency and security. While local load balancing happens within a single data center, geographic load balancing uses multiple data centers in many locations.
- **Multi-site** — Multi-site load balancing, also known as global server load balancing (GSLB), distributes traffic across servers located in multiple sites or locations around the world. The servers can be on-premises or hosted in a public or private cloud. Multi-site load balancing is important for quick disaster recovery and business continuity after a disaster in one location renders a server inoperable.
- **Load Balancer as a Service (LBaaS)** — Load Balancer as a Service (LBaaS) uses advances in load balancing technology to meet the agility and application traffic demands of organizations implementing private cloud infrastructure. Using an as-a-service model, LBaaS creates a simple model for application teams to spin up load balancers.