Application Delivery Controller | White Paper

What is an Application Delivery Controller (ADC)?

citrix.com

ADCs are advanced load balancers with functions and features that enhance the performance of applications. Today, companies of all sizes with geographical dispersal of people and different data constructs require ADCs to optimize their complex application environments from web applications, to Exchange, SharePoint and databases. It is interesting that before the term ADC was used more recently (in the last decade), companies relied on load balancers for website availability and scalability. In this paper we will describe the fundamentals of a load balancing system and its evolution to an ADC.

Load balancers

Load balancing is a networking method to distribute workload across multiple servers to achieve optimal resource utilization, maximize throughput, minimize response time, and avoid overload. The load balancing service was usually provided by dedicated software or hardware, such as a multilayer switch or a Domain Name System (DNS) server. Using multiple servers with load balancing, instead of a single server, may increase reliability through redundancy.

Round-robin DNS

One of the first methods of load balancing was round-robin DNS. In this technique, multiple IP addresses are associated with a single domain name; the domain name system decides which server address to give to the connecting client. This technique exposes to clients the existence of multiple backend servers. The technique has advantages and disadvantages, depending on the degree of control over the DNS server and the granularity of load balancing desired.

Load balancing is dividing the amount of work that a server has to do between two or more servers so that more work gets done in the same amount of time and, in general, all users get served faster. Load balancing can be implemented with hardware, software, or a combination of both. For Internet services, the load balancer is a software program that is listening on the network port where external clients connect to access services. The load balancer chooses the best “back end” server to handle the request and then forwards it. The “back end” server usually replies to the load balancer. This allows the load balancer in turn to reply to the client without revealing the internal separation of functions. It also prevents clients from contacting the back end servers directly, providing an additional layer of security by concealing the structure of the internal network and preventing direct attacks to the servers.

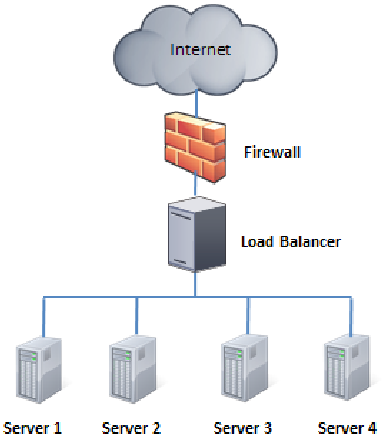

Figure 1. A simple load balancer

Some load balancers provide a mechanism for taking action in the event that all back end servers are unavailable. This might include forwarding requests to a backup load balancer, or displaying a message regarding the outage. Load balancing gives the IT team a chance to achieve significantly higher fault tolerance for their services. It can also automatically provide the amount of capacity needed to respond to any increase or decrease of application traffic.

TCP load balancing

TCP/IP load balancing is a mechanism for distributing many incoming service requests between two or more servers. Unlike with DNS round-robin, all clients attach to a single IP address. The distribution is adjusted by a load balancing algorithm specific to the service. This improves and optimizes service response times, service reliability, and the overall user experience.

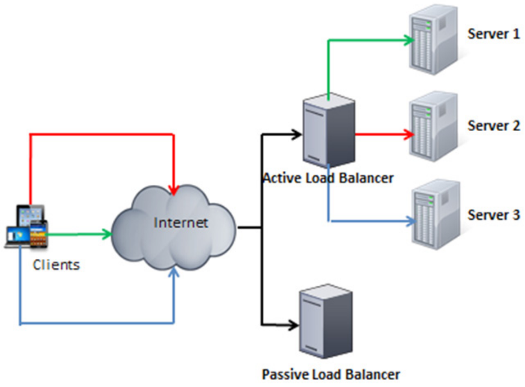

Figure 2. Example of a distributed client load with a web server farm

Two or more servers are grouped together in a single configuration that is identified by a single host name or TCP/IP network address. Any application or client requiring a service sends the request to the single host name that is hosted on a load balancer. The load balancer receives the requests and selects which one of the servers within the group will perform the task.

When a load balancer selects the appropriate server to satisfy the client request, it uses a load balancing algorithm. A variety of scheduling algorithms are used by load balancers to determine which back end server to send a request to. Simple algorithms include random choice or round robin. More sophisticated load balancers take into account additional factors, such as a server's reported load, recent response times, up/down status (determined by a monitoring poll), number of active connections, geographic location, capabilities, or how much traffic it has recently been assigned.

Emergence of Application Delivery Controllers

In late 2000s, web servers weren't just delivering static content, they were delivering applications. Businesses were using web based applications to deliver mission critical functionality to employees, customers, partners and contractors. Simple server load balancing was no longer sufficient as these mission critical applications delivered dynamic and real-time content. Users needed to be connected to application servers based on a variety of criteria using policies and advanced application-layer knowledge to support business requirements. To accommodate many internal applications without making changes to them directly, an ADC can transform or rewrite the content of the client request including the response from the servers.

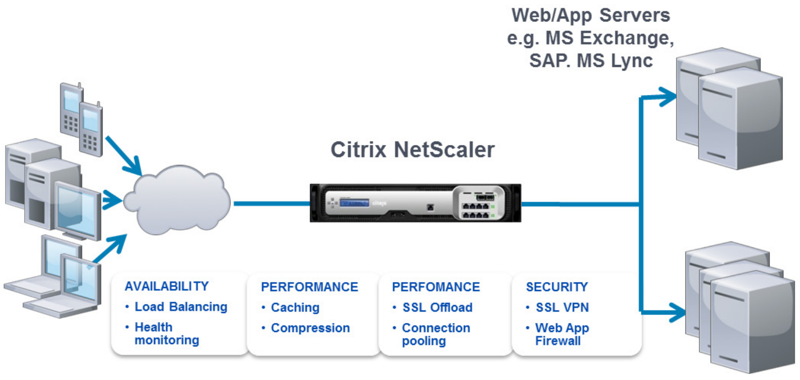

Figure 3. An ADC optimizes application workloads

An ADC is typically placed in a datacenter between the firewall and one or more application servers. For users requesting applications and content from a datacenter, an ADC will route users to destination servers based on a variety of criteria that the datacenter manager implements using policies and advanced application-layer knowledge to support business requirements. An ADC will ensure that the users get to the applications based on their specific needs while protecting the network and applications from security threats using an application firewall. ADCs understand how applications work and can look deeper into the specific traffic and make more intelligent decisions. They can optimize application server performance by offloading many compute-intensive tasks that would otherwise bottleneck the server CPUs needed to deliver applications to users. Some additional functions commonly present in ADCs are SSL offload technology, accelerated compression, TCP and HTTP protocol optimization and virtualization awareness.

ADCs offer compression to help deliver maximum bandwidth utilization thus supporting more traffic and avoiding the need for network upgrades. Currently, ADCs are being enhanced by the addition of new protocol intelligence and new services, such as content transformation and transaction assurance. An ADC includes many features. Some of the main features are listed below:

- **Asymmetric load** – A ratio can be manually assigned to cause some back end servers to get a greater share of the workload than others. This is sometimes used as a rudimentary way to account for some servers having more capacity than others.
- **Priority activation** – When server availability drops below a certain number, or load gets too high, standby servers can be brought online. This helps ensure that resources are used efficiently within the datacenter.
- **SSL Offload and Acceleration** – Processing the encryption and authentication requirements of a Secure Socket Layer (SSL) request can become a major part of the demand on the Web Server's CPU and as the demand increases the users will experience slower response times. To remove this demand from the Web Server an ADC may be used to terminate the SSL. Some ADC appliances include specialized hardware to process SSL for application performance gains.
- **Distributed Denial of Service (DDoS) attack protection** – Distributed denial-of-service attack (DDoS attack) is an attempt to make a machine or network resource unavailable to its intended users. Using its protocol analysis and content switching features, the ADC can direct flows to various servers based on type and requested content. These features can be configured to queue the most time-consuming and least critical requests or to reject improperly formed http requests or requests that refer to elements that do not exist. By stepping in between attackers and the servers, the ADC can block undesirable requests while maintaining traffic to critical applications.
- **HTTP compression** – Reduces amount of data to be transferred for HTTP objects by utilizing Gzip and Deflate compression available in all modern web

browsers. The larger the response and the further away the client is the more this feature can improve response times by reducing the number of round-trips required to retrieve the data.

- **TCP offload** – Different vendors use different terms for this, but the idea is that normally each HTTP request from each client is a different TCP connection. This feature utilizes HTTP to consolidate multiple HTTP requests from multiple clients into a single TCP socket to the back-end servers.
- **TCP buffering** – The ADC can buffer responses from the server and drip-feed the data out to slow clients, allowing the web server to free a thread for other tasks faster than it would if it had to send the entire request to the client directly.
- **Health checking** – The ADC will poll servers for application layer health and remove failed servers from the pool resulting in high availability.
- **HTTP caching** – The ADC can store static content so that some requests can be handled without contacting the web servers. This results in performance gains by speeding up the response from a busy server as well as reducing the number of servers required to deliver content.
- **Content filtering** – Load balancers can intelligently modify traffic on the way through as it provides protection from malicious attacks.
- **HTTP security** – ADCs can hide HTTP error pages, remove server identification headers from HTTP responses, and encrypt cookies so end users can't manipulate them.
- **Priority queuing** – Also known as rate shaping, the ability to give different priority to different traffic.
- **Content-aware switching** – ADCs can send requests to different servers based on the URL being requested, assuming the request is not encrypted (HTTP) or if it is encrypted (via HTTPS) terminate the HTTPS request (decrypted) at the ADC and apply policies.
- **Client authentication** – Authenticate users against a variety of authentication sources before allowing them access to a website.

Citrix NetScaler, the industry-leading application delivery controller with advanced load balancing is deployed in thousands of networks around the globe to optimize, secure and control the delivery of all enterprise and cloud services. NetScaler load balancing ensures that applications and services are 100% available to all users. Application performance is maximized by a set of powerful acceleration capabilities, including intelligent data compression, static and dynamic content caching, and multiple TCP optimizations that improve the efficiency of the network. The NetScaler App Firewall protects against web application-layer attacks.

For more information on Citrix NetScaler Application Delivery Controller please go to: http://www.citrix.com/products/netscaler-application-delivery-controller/overview.html

Conclusion

Since applications are the lifeline of any business it is imperative that they are available, reliable and secure 24/7. This is where ADCs play a crucial role by providing a set of functions to optimize enterprise application environments. Enterprises use ADCs to optimize reliability, end-user performance, datacenter resource use and security for a variety of enterprise applications. The market evolved from the load-balancing systems that were specifically developed to ensure the availability and scalability of websites.

Corporate Headquarters
Fort Lauderdale, FL, USA

Silicon Valley Headquarters
Santa Clara, CA, USA

EMEA Headquarters
Schaffhausen, Switzerland

India Development Center
Bangalore, India

Online Division Headquarters
Santa Barbara, CA, USA

Pacific Headquarters
Hong Kong, China

Latin America Headquarters
Coral Gables, FL, USA

UK Development Center
Chalfont, United Kingdom

About Citrix
Citrix (NASDAQ:CTXS) is the cloud company that enables mobile workstyles—empowering people to work and collaborate from anywhere, easily and securely. With market-leading solutions for mobility, desktop virtualization, cloud networking, cloud platforms, collaboration and data sharing, Citrix helps organizations achieve the speed and agility necessary to succeed in a mobile and dynamic world. Citrix products are in use at more than 260,000 organizations and by over 100 million users globally. Annual revenue in 2012 was $2.59 billion. Learn more at www.citrix.com.