White Paper

A Guide to Application delivery Optimization and Server Load Balancing for the SMB Market

KEMP
TECHNOLOGIES, INC.

Introduction

Today's small-to-medium sized businesses (SMB) are undergoing the same IT evolution as their enterprise counterparts, only on a smaller scale. For SMBs, website reliability, flexible scalability, performance and ease of management are as essential to SMB website infrastructure as they are to an enterprise. It's fare to say that these capabilities are an important operational imperative for businesses of all sizes. SMBs can gain efficiencies and competitive advantages by adopting appropriate networking technologies. However, without the proper systems in place, they will suffer from poor performance and they will be competitively disadvantaged. For this reason, choosing the appropriate application delivery controllers and server load balancing products is critical to ensuring efficient and effective website infrastructure to meet today's needs, while ensuring the right upgrade path for tomorrow's business requirements.

SMB application delivery and server load balancing needs

Until recently, even basic server load balancing was cost prohibitive for SMBs. Today, advanced application delivery controllers and intelligent load balancers are not only affordable, but the consolidation of Layer 4-7 load balancing and content switching, and server offload capabilities such as SSL, data caching and compression provides SMBs with cost-effective out-of-the-box infrastructure.

For enterprise organizations (companies with 1,000 or more employees), integrating best-of-breed network infrastructure is commonplace. However, for SMBs, best-of-breed does not equate with deploying networks with enterprise-specific features and expensive products, but rather, deploying products that are purpose-built, with the explicit features, performance, reliability and scalability created specifically for the SMB market.

In general, businesses of all sizes are inclined to purchase "big brand" products. However, smaller vendors that offer products within the same category can provide the optimal performance, features and reliability that SMBs require, with the same benefits - at a lower cost.

For the enterprise market, best-of-breed comes with a high Total Cost of Ownership (TCO), since deploying products from various manufacturers requires additional training, maintenance and support. KEMP can help SMBs lower their TCO, and help them build reliable, high performance and scalable web and application infrastructure. KEMP products have a high price/performance value for SMBs. Our products are purpose-built for SMB businesses for dramatically less than the price of "big name" ADC and SLB vendors who are developing features that enterprise customers might use.

Proven solutions for cost-effective, reliable application delivery and server load balancing

Other vendors offer application delivery controllers and server load balancers, but KEMP products have the best price/performance value as determined by rich features, scalability, high-availability, performance and ease of management – at a cost-effective price to meet the needs of small- to-medium size businesses.

Web applications

Up until the last few years, businesses typically had separate systems and services to communicate and transact business with customers, partners and employees. Now, through the ubiquitous acceptance and accessibility of the Internet, the real power of networking is being utilized. Traditional applications such as order processing, billing and customer management have been integrated into complete supply-chain web applications. These new web applications now unify and streamline business processes from previously monolithic client/server business applications. This is good news for small-to-medium sized businesses (SMB), as web-based applications offer the potential to reduce the need for expensive hardware, reduce time-to-market and lower maintenance costs.

It’s no cakewalk

Many organizations that deploy web-based applications face a myriad of challenges - from initial deployment to production. For example, you may run into problems when your servers cannot handle the number of visitors to your site. The sources of problems are invariably due to high traffic volumes connecting to the network, and limitations due to server resources. Despite increasing budgets for servers and network upgrades, web applications may not deliver the expected improvements in performance, scalability and efficiency.

IT Infrastructure

E-commerce is conducted by using transaction-based websites that can be highly complex and expensive to manage. A typical e-commerce website is connected to routers that pass traffic through firewalls, which pass traffic to application delivery controllers (the next generation server load balancers) that ultimately direct users to the appropriate servers. Network and application delivery optimization products distribute the traffic to many diverse servers that are often connected to database servers. If just one of these components in the process fails, a worst case scenario would be that the entire site can be taken down. Often, what happens is a user request will be delayed, or a customer transaction will fail.

The Internet is a highly resilient network. However, it was not developed with the demands of modern commerce in mind. With today’s use of the Internet, a moment's delay can cost a business thousands, or even millions of dollars. Even though new web-based applications are designed with this in mind, both the Internet and the server resources can be a bottleneck. The Internet does not distinguish between a business-critical transaction and a benign web page, and does not assign guaranteed quality of service for applications. If you had an unlimited IT budget for servers, systems, bandwidth and personnel to manage and monitor your website infrastructure, you might be ok, but for virtually all organizations, that is unreasonable.

Application delivery solutions

Application delivery solutions were built to address the challenges associated with website infrastructure complexity, performance, scalability and security. Application delivery solutions are quite diverse. They may be known as application delivery controllers (ADC), application delivery controllers, server load balancers (SLB), application front-end devices (AFE), application traffic managers, and web front-ends, content switches and application switches. In order to avoid confusion, this paper will focus on datacenter solutions, and refer to application delivery solutions as application delivery controllers. Today’s application delivery controllers actually evolved from server load balancers that were first introduced in the late 1990s.

ADCs provide the ability to direct Internet users to the best performing, most accessible servers. Should one of the servers (or applications on that server) become inaccessible due to any type of failure, the ADC will take that server or application off-line, while automatically re-routing users to other functioning servers. This process is essentially seamless to the user, and critical to servicing the customer.

During the past five years, application delivery has emerged as one of the most important technologies in solving the problem of performance and accessibility for Internet-based applications.

In addition, by using various load balancing algorithms, an ADC can distribute users to servers that offer the best possible performance. The ADC can dynamically interrogate key server elements such as the number of concurrent connections and CPU/memory utilization.

To further enhance, and secure the user experience, more-advanced ADCs provide SSL offload/acceleration. SSL acceleration in the ADC enables you to offload the SSL handshake and encryption/decryption processes from the servers. This offloading dramatically increases the servers' performance, while decreasing the time and costs associated with the server's SSL certificate management.

Significant demand in the marketplace will come from the integration of Web 2.0 applications. This demand will be satisfied by ADCs. In addition to SSL acceleration, there are three primary elements to application acceleration.

- Content caching - stores data that is likely to be used again and is unlikely to change, rather than requiring computers to retrieve it from the source every time.
- Data compression - reduces the amount of data crossing the link — squeezing the data into smaller packets, which are then combined into a larger packet — making it faster and more efficient to send across a network.

Application delivery controllers use various techniques to distribute the traffic load between two or more servers, routers, firewalls, and other networked resources, to optimize resource utilization and improve website performance and response time.

Most application delivery controllers are capable of providing Layer 4 to Layer 7 management. Layer 4 is limited to web requests destined to TCP Port 80; therefore, no further differentiation among server groups is possible. However, Layer 7 switching uses application-layer criteria to determine where to send a request. This provides an application delivery controller with much more granular control over forwarding decisions. In diagram 2 below, using the HTTP protocol, www.xyz.com/images may be pointed to a different server(s), while the www.xyz/shop might be directed to image servers.

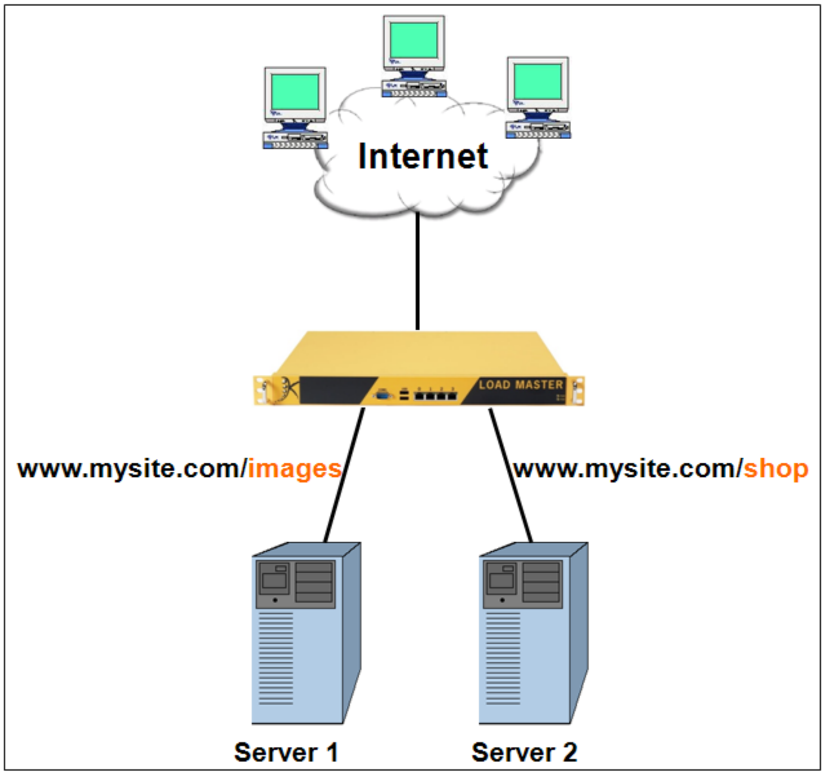

Diagram 2: Redirect users to appropriate servers based on Layer 7 switching.

Another unique use of Layer 7 for persistence is maintaining user persistence to Microsoft Windows Terminal Services (WTS).

What to look for in an application delivery controller

The functionality listed below is the criteria that a small-to-medium sized organization should look for when choosing a network and application delivery solution.

High Availability (Hot Standby)

Since all inbound traffic must pass through the application delivery controller, should it fail, the server farm and the entire site will not be accessible. To address this, most vendors support redundant configurations. Usually, a standby (or redundant) configuration is supported - sometimes referred to as HA (High Availability). Most sites utilize at least one HA pair - as it would be risky to deploy multiple web servers for redundancy and scalability, only to lose the entire site due to the ADC hardware failure. Should one of the LoadMaster products goes down, the standby unit will send an SMNP notification to the administrator.

Layer 4 Load Balancing

Application delivery controllers provide for multiple Layer 4 IP-based methods for distributing user traffic to servers. These methods may include Round Robin, Weighted Round Robin, Least Connection, Weighted Least Connection, Chained Failover (Fixed Weighting), and Server-Resource-Based. Weighting enables the administrator to assign a higher (or lower) "weight" to real servers, so as to provide better control over traffic distribution. For instance, if two web servers are based on the Intel P3 platform and you add a new quad-core, dual Xeon-based server with 16GB ram, you can designate the new server with a higher "weight", thereby sending 2, 3 or 100 times more traffic to the new server.

While Layer 4 load balancing methods are sufficient for many low-end, low volume Internet traffic, a large number of sites (and rapidly growing) require a much more granular approach to traffic distribution, which can only be accomplished through higher level load balancing methods, such as Layer 7 content switching.

Layer 7 Content Switching

Content switching refers to the ability to distribute (or load balance) user requests to servers based on Layer 7 payload. Most commonly, this is done by examining page content (such as a URL) and "switching" the requests to the appropriate server or group of servers. As an example, www.xyz.com/images may be pointed to a server that handles graphics, while the www.xyz/shop may be pointed to a transaction server. This provides for much greater performance tuning and application flexibility. In addition, if your web application is making extensive use of cookies, a Layer 7-capable application delivery controller can switch users based on cookie values, providing a much better model for achieving server persistence.

IP Persistence (Layer 4 Persistence)

"Persistence" (sometimes referred to as "Sticky" or server "Affinity") is best understood by looking at the example of the "shopping-cart". The shopping-cart, as employed by most e-commerce sites, is a logical repository for items that have been selected by a client while shopping at an online site. The items selected generally reside on the server to which the client first connected, and which served the client the content during the session. If at any point during the session the client is switched to another server (a server that does not share the session-data), the client's shopping-cart will be lost, and the shopping experience will be ruined. Persistence is a method of ensuring that for a prescribed duration, the user always comes back to the server where their data is located.

There are a number of ways that persistence can be accomplished. The persistence mechanism was first employed by load-balancers, based on Source IP Address. Using Source IP, load-balancers attempted to identify users by their Source IP Address, and to keep the users "stuck" to the appropriate server using this identifier. This method quickly proved to be unreliable due to the effect of proxy servers and network-address translation (NAT). When proxy-servers or NAT are used, there is no way to reliably correlate an IP address to a user. Instead, many users may be represented by a single IP address, or a single user's IP address may change throughout the life of a session. Because of this pervasive effect, Layer 7 (or Cookie) persistence is much more reliable, and is most commonly offered by application delivery controllers.

Layer 7 Persistence (e.g. Cookie Persistence)

Advanced Layer 7 application delivery controllers offer the ability to inspect the data at the application layer. With this, comes a persistence mechanism known as Cookie Persistence. Cookie Persistence uses a browser cookie to uniquely identify users. Either the application or the application delivery controller itself can serve cookies to users at the start of a session, and the user's browser can automatically return the cookie during each successive hit. By tracking this cookie information, the application delivery controller is able to accurately determine which server should receive the subsequent request. Cookie persistence remains the most reliable method of achieving persistence.

SSL Acceleration

If your site contains "transactional" elements, chances are that all or some portion of your site uses SSL to encrypt and secure those transactions. In years past, SSL was handled by the same server that served content. However, since SSL processing takes a significant toll on the overall server performance, server-side SSL handling was less then optimal to say the least. Several options exist in aiding of SSL processing at the server level, including installation of SSL acceleration cards that are designed to offload the CPU-intensive SSL handshaking and session setup and teardown, but this solution introduced other complexities, and still did nothing for aiding Layer 7 content switching and persistence.

It became clear that the best place to offload SSL processing was at the application delivery controller - not at the server. Once SSL was moved to the application delivery controller, all HTTPS requests would terminate at the application delivery controller, get decrypted and routed to the appropriate real server via HTTP. This offered a number of advantages including SSL certificate aggregation (now you would only need one certificate which resides on the LoadMaster, rather than one certificate per server), but mainly it allowed for the decryption of content - therefore providing the application delivery controller with the ability to perform content switching and layer 7 persistence on traffic that originated as HTTPS.

Using Persistence with SSL

Unable to rely upon Source IP addresses, and unable to inspect the cookie during an SSL session, application delivery controller vendors had to devise a method of offering persistence in an SSL environment. Fortunately, SSLv3 offers a way to do this. SSLv3 moved the SSL Session ID, a unique 32 byte session identifier, out of the encrypted portion of the data into a clear portion. Application delivery controller are able to read this unique identifier, and balance the traffic to the appropriate server based on this predictable identifier. This method worked well until Microsoft introduced Internet Explorer version 5.0. Beginning with IE5, Microsoft changed the SSL behavior to force a renegotiation of a new SSL session every two minutes. This meant that all IE5 and newer users would change SSL Session ID every two minutes, breaking the only method of secure persistence available.

By coupling an SSL accelerator with an application delivery controller, it once again became possible to offer reliable persistence by decrypting the SSL content so that the application delivery controller can inspect the data again. When SSL traffic arrives at the application delivery controller, it is redirected to an SSL offload function. The SSL accelerator decrypts the content in its entirety, including any cookies that might have been sent by the browser, and sends them back to the application delivery controller. At this point, the data is in-the-clear, and the application delivery controller can inspect the cookies, the URL, the URI, the browser type, etc.

Windows Terminal Services Load Balancing, Persistence and Session Directory Management

Application Delivery Controllers can provide full support for Windows Terminal Services 2003 and Windows Server 2008. ADC resource monitoring provides data on both server memory and CPU, ensuring users experience the most efficient load balancing possible across each server. ADC integrates integration with WTS Session Directory can provide a reliable "re-connect" when a remote desktop connection to the server has disconnected. An ADC that supports RDP-based Layer 7 persistence can incorporate client session reconnect, which can be utilized without the need for the Session Directory service to be installed. This helps simplify IT infrastructure, and provides cost savings benefits.

Number of LAN Ports

It is important to note that the total number of ports has nothing to do with the number of servers that the application delivery controller can support. Using your existing layer 2/3 switch or hub, you can support many more servers then the number of actual ports provisioned on the application delivery controller. In fact, using what is sometimes referred to as a "single arm" configuration, you can load balance up to 1000 real servers on a single application delivery controller.

Maximum Real Servers

Maximum Real Servers refers to the number of physical servers that an application delivery controller can support. Depending on the ADC model, you will have a limitation of the number of real servers that can be supported by one virtual server (VIP). However, you are much more likely to exceed the capacity of your Internet connection well before you reach the limit of how many servers the application delivery controller can support.

An application delivery controller consists of a virtual server (also referred to as a virtual cluster, virtual IP or VIP) which, in turn, consists of an IP address and port. This virtual server is bound to a number of physical (real) servers within a server farm. On the application delivery controller, a virtual server (VIP) is typically a publicly facing IP address which responds to user requests. Typically, load balancing, content switching and persistence rules and methods are assigned on a per-VIP basis. In the case of real servers, having a good number of supported VIPs presents more flexibility in the architecture and design of the site or application - since multiple VIPs can be pointed to the same set of real servers. A robust application delivery controller will support up to 256 VIPs.

Maximum Real Servers per VIP

Most application delivery controller vendors set a limit on how many real servers can be addressed by a single VIP. Too small of a ratio of VIP to real servers may inhibit the design and architecture in certain applications. A good application delivery controller should support up to 100 real servers per VIP.

Throughput

While most venders provide the theoretical maximum bandwidth capacity of the Ethernet interface, this number has very little to do with the actual throughput of the application delivery controllers, since this number is highly dependent on the number and type of rules that the application delivery controller has to analyze, as it decides on how to deal with the packet. For example, if the application delivery controller has to make a load balancing decision based on Layer 7 content, the additional latency associated with this process may have a significant impact on overall performance and total throughput. A quality application delivery controller appliance will use modern hardware architecture to allow it to sustain excellent Layer 7 performance.

Maximum SSL TPS

TPS refers to the number of SSL transactions that the application delivery controller can handle per second. An SSL transaction includes some fairly intensive number crunching associated with SSL key exchange and the setup and teardown of the SSL connection. An application delivery controller that includes an ASIC to offload SSL processing from the main CPU will deliver greater performance. Typically, ASIC cards are quite expensive and most vendors often charge extra fees for licenses for additional TPS and/or for the addition of SSL ASIC. It is important to find a vendor where the list price already reflects maximum SSL TPS. This will ensure that there are no additional fees for TPS licensing or co-processor boards. However, keep in mind that at the low-end of the SSL performance, the key advantage to having SSL offloading at the application delivery controller is the ability to decrypt HTTPS packets, and make balancing and persistence decisions at Layer 7.

Direct Server Return (DSR)

In some applications it may be desirable (or required) for the real server to respond to the client requests directly – without having to go through the application delivery controller. The main benefit of using Direct Server Return (DSR) is that by bypassing the application delivery controller, the servers can transfer large payloads of data (such as streaming videos, large page loads, file transfers, etc.) directly to the client, avoiding any latency that may be associated with an application delivery controller – and therefore, increase the performance of the application and site. While an application delivery controller may support DSR, it should be noted that since DSR is a Layer 4 function, any Layer 7 features (such as cookie persistence) will not be available in this configuration.

Transparency

An application delivery controller is often configured as a “NAT” device. This implies that when the application delivery controller is communicating with a real server, the “client” IP address that is presented to the server is that of the application delivery controller, and not of the actual user making a request. In some environments that may present a problem, since the original client’s IP address is not present in the server’s logs. To address this, some application delivery controllers provide a feature called “Transparency” – which provides administrators with a way to preserve a client’s IP address in their server logs. There are some important architectural trade-offs that need to be made if transparency is required, so please consult the application delivery controller manual, or contact the vendor for assistance.

HTTP Compression

HTTP compression reduces the amount of data to be transferred for HTTP objects by utilizing gzip compression available in all modern web browsers. HTTP compression allows application delivery controllers to compress the application payload within each packet. Overall, it reduces network bandwidth consumption without degrading content quality, and improves the end-users' overall experience. HTTP compression running on an application delivery controller offloads this processor-intensive task from servers.

HTTP Caching

Application delivery controllers with caching capabilities serve as proxy caches, storing selected data from origin servers to speed delivery to clients. The combination of caching and compression can deliver a double benefit, because devices can return pre-compressed objects out of cache, rather than retrieving them from origin servers. Caching can substantially boost transaction rates and reduce response times, while also freeing servers to do other work. Chatty protocols such as HTTP require frequent creating and tearing down of connections, creating unnecessary resource utilization on servers. Application delivery controllers with caching capabilities enable you to repurpose connection related resources for more relevant business logic.

Intrusion Prevention

For enhanced security, an application delivery controller with Intrusion Prevention System (IPS) provides in-line protection of bandwidth and servers by enabling real-time mitigation of attacks and isolation of servers. Intrusion prevention provides real-time intrusion alerting.

Resource-based Load Balancing

Resource-based load balancing allows the use of a scripting language to provide custom load balancing methods, arbitrary traffic manipulations, and more.

Summary

The complexity associated with the technology required to manage web infrastructure within small-to-medium sized businesses and managed hosting services has brought with it many new challenges that today's IT staff must meet. The new and varied technologies implemented within datacenters is usually more than just a single web server, and can often exceed the management, performance and scalability needs that these organizations require. Compound this with the fact that immediate and secure access has never been more of a concern, yet many websites today still lack the required web infrastructure to deliver the appropriate reliability, performance and security.

For SMB and managed hosting providers, the complexity and dynamic nature of e-commerce are the major causes of poor site performance and unplanned downtime. SMBs and service providers are becoming increasingly aware of the need to protect these vital, yet vulnerable sites. However, acquiring more devices, more complexity and more single capability solutions is not the answer. Optimizing the delivery of applications between end-users and diverse datacenter equipment, by providing ease of management, faster access to applications and content and security within a cohesive platform is required.