Progress®
Kemp® LoadMaster®

Product Overview Progress Kemp LoadMaster

Generated on: 21 April 2026

Copyright

Visit the following page online to see Progress Software Corporation's current Product Documentation Copyright Notice/Trademark Legend: Product Documentation Copyright Notice & Trademarks | Progress

Table of Contents

Chapter 1: Introduction to Progress Kemp and the LoadMaster Products 7

LoadMaster Load Balancer Features 8

Chapter 2: LoadMaster Network Topologies 9

One-Armed Balancer. 10

Two-Armed Balancer. 12

High Availability (HA) Configuration 14

Clustering 18

Direct Server Return – DSR Configuration Example. 21

Chapter 3: Scheduling Methods 24

Chapter 4: Persistence. 29

How Do I Know If I Need Persistence? 32

Timeout 32

Layer 7 Persistence Methods 33

Persistence and HTTPS/SSL 36

Port Following 36

Chapter 5: Application Front End 38

Intrusion Prevention System 39

Intrusion Handling. 40

Detection level 40

Warnings 40
Intrusion Alerts 40
Updating IPS Rules 41
Caching 41
Data Compression 42

Chapter 6: SSL Acceleration/Offloading 44

Chapter 7: Edge Security Pack (ESP) 46

Chapter 8: Web Application Firewall Pack (WAF) 51
Challenges with Traditional Firewalls 51
Benefits of WAF 52
WAF-enabled LoadMaster Overview 52

Chapter 9: GEO 55

Chapter 10: Sub-Virtual Services (SubVSs) 56

Chapter 11: Certificates 58
Self-Signed Versus CA-Signed Certificates 58
Certificate Basics 59
Operational Differences 59
ACME Certificates 60
Let's Encrypt Certificates 60

DigiCert Certificates 61

Chapter 12: Rule-Based Content Switching 62

Chapter 13: Health Checking. 65

Overview 65

Service and Non-Service Based Health Checking 67

Chapter 14: SNMP Support 71

Chapter 15: LoadMaster Software Upgrades. 73

Chapter 16: User Management 75

Chapter 17: WUI Authentication and Authorization 77

Chapter 18: Bonding and VLAN 79

Prerequisite (Switch Compatibility) 79

Bonding/Teaming (802.3ad/Active-Backup) 80

VLAN Tagging 80

Chapter 19: IPsec Tunneling 81

Chapter 20: Miscellaneous 82

IPv6 Support 82

Remote Syslog Support 83

How to Get a License 83

Backup and Restore 84

Disable/Enable Access to the WUI 85

Interoperability Between L4 and L7 Virtual Services 85

Log Information 85

Debugging Utilities 86

RESTful API Interface 86

Chapter 21: Network Telemetry 87

Chapter 22: References 88

1

Introduction to Progress Kemp and the LoadMaster Products

Refer to the sections below for an introduction to Progress Kemp, the LoadMaster products, and their features.

Progress Kemp

Progress Kemp leads the industry in driving the price/performance value proposition for application delivery and load balancing to levels that our customers can afford. Our products' versatile and powerful architecture provide the highest value, while enabling our customers to optimize their businesses that rely on Internet-based infrastructure to conduct business with their customers, employees and partners.

The LoadMaster Products

The LoadMaster family of feature-rich application delivery controllers and server load balancer appliances automatically and intelligently manage user traffic and applications, to deliver website integrity for all sizes of businesses and managed service providers.Progress Kemp products optimize web infrastructure as defined by high-availability, high-performance, flexible scalability, ease of management and secure operations - while streamlining IT costs. LoadMaster simplifies the management of networked resources, and optimizes and accelerates user access to diverse servers, content and transaction-based systems.For a lot of organisations, having the company website or intranet accessible, secure and continuously operating is critical. With a powerful Application Delivery Controller (ADC) or load balancer from Progress Kemp, you will be providing your business with a high-value, reliable infrastructure appliance that will significantly improve your web server performance, reduce costs and increase your customer's web experience.

Contents

- LoadMaster Load Balancer Features

LoadMaster Load Balancer Features

The LoadMaster Application Delivery Controller (ADC) provides the following features with the LoadMaster Operating Software and the Web User Interface (WUI):

- Balancing Methods
- Persistence
- Application Front End
- SSL Acceleration/Offloading
- Rule Based Content Switching
- Health Checking
- GEO
- Web Application Firewall (WAF)
- SNMP Support
- User Management
- IPv6 Support
- Bonding and VLAN
- Edge Security

2

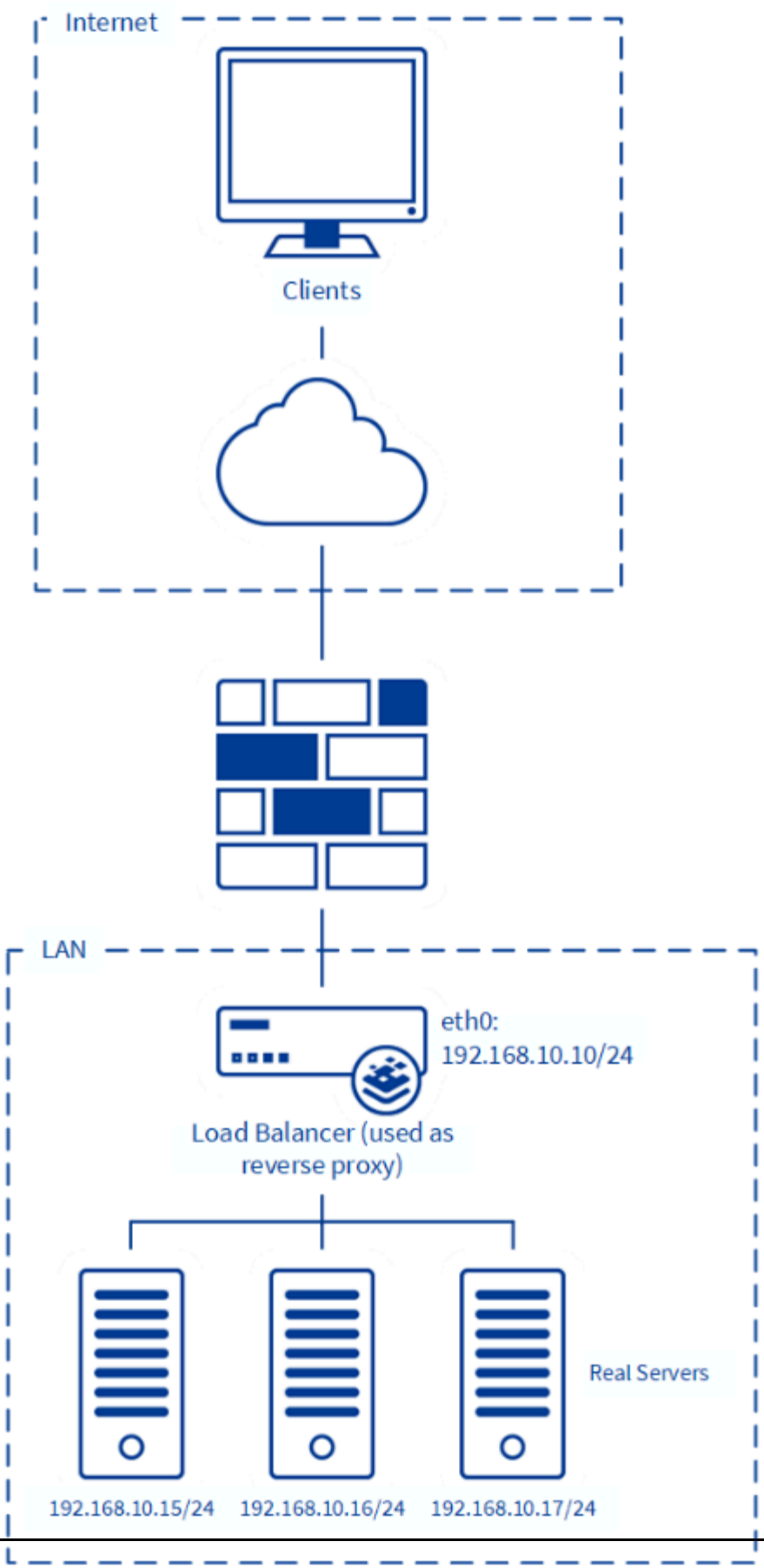
LoadMaster Network Topologies

Refer to the following sections for an overview of various network topologies.

Contents

- One-Armed Balancer
- Two-Armed Balancer
- High Availability (HA) Configuration
- Clustering
- Direct Server Return – DSR Configuration Example

One-Armed Balancer


Internet
Clients
LAN
eth0:
192.168.10.10/24
Load Balancer (used as
reverse proxy)
Real Servers
192.168.10.15/24
192.168.10.16/24
192.168.10.17/24

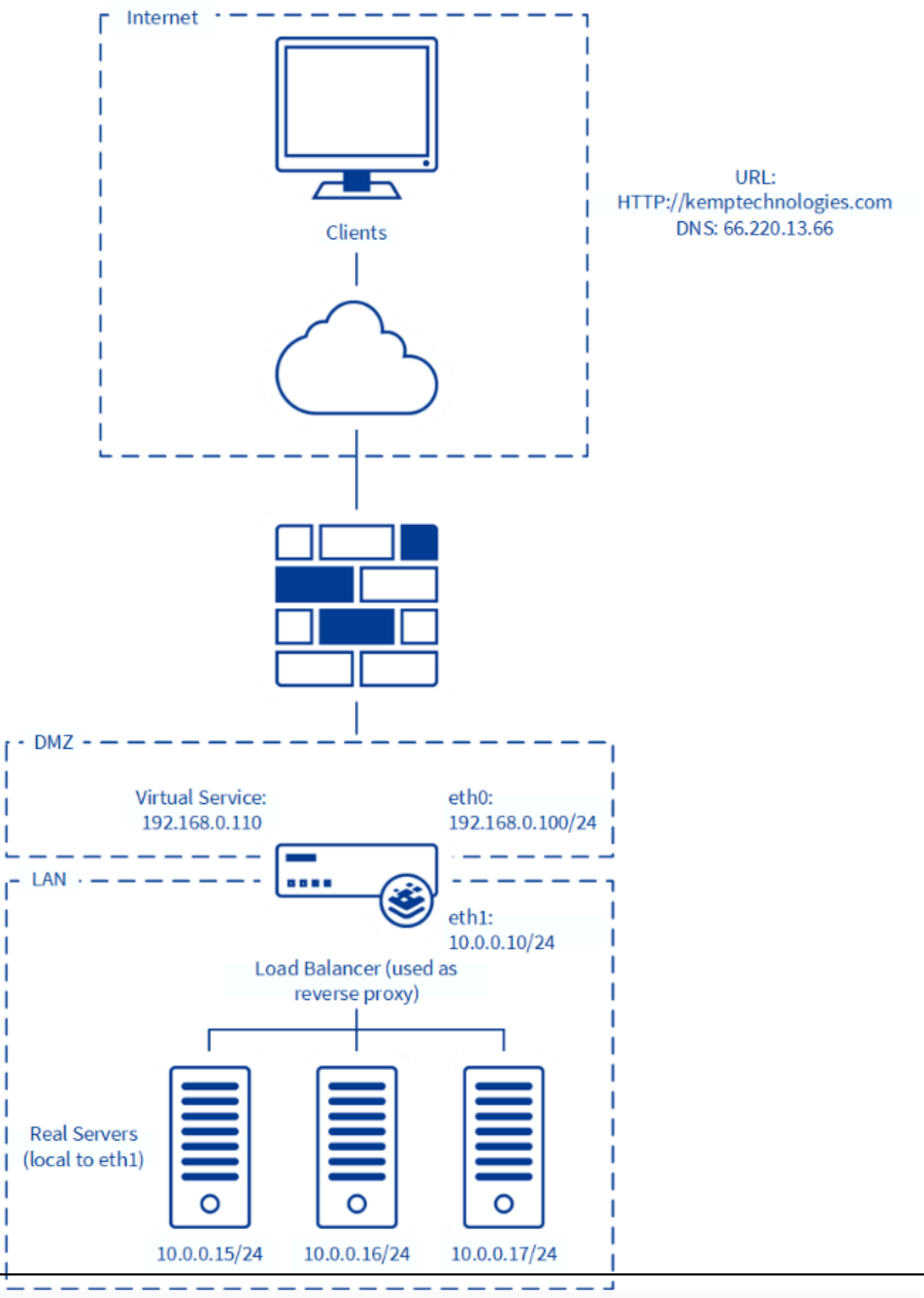
If a one-armed configuration is selected, then the following is true:

- Real Servers and Virtual Services will be part of the same logical network – sometimes called flat-based - this implies that both have public IP addresses if used for services within the Internet.
- Server NAT does not make sense for one-armed configurations.
- Does not automatically imply the use of Direct Server Return (DSR) methods on the Real Servers
- IP address transparency will function properly if clients are located on the same logical network as the LoadMaster in a DSR configuration. IP address transparency is not supported when clients are located on the same logical network as the LoadMaster in a NAT configuration. Refer to the Transparency Feature Description for further details on transparency.

The one armed solution may be set-up in both a single and HA configuration.

Two-Armed Balancer

An example of a two-armed LoadMaster site may look as follows.


Internet
Clients
URL:
HTTP://kemptechnologies.com
DNS: 66.220.13.66
DMZ
Virtual Service:
192.168.0.110
eth0:
192.168.0.100/24
LAN
eth1:
10.0.0.10/24
Load Balancer (used as
reverse proxy)
Real Servers
(local to eth1)
10.0.0.15/24
10.0.0.16/24
10.0.0.17/24

The system has been configured as follows:

- A Virtual Service has been created on the LoadMaster with an IP address of 192.168.0.110 for an HTTP service.
- The Virtual Service has been configured to balance the incoming traffic across the Real Servers (server 1, 2 and 3).
- A user requests the URL http://www.kemptechnologies.com.
- The URL will be resolved by the DNS into IP address 66.220.13.66.
- The request will be routed to the firewall, which will NAT to the Virtual Service address (192.168.0.110) on eth0 (192.168.0.100/24).
- The LoadMaster is connected to the server farm subnet 10.0.0.0/24 using its network interface eth1 (10.0.0.10/24).
- The LoadMaster knows that there are three Real Servers in this subnet that are assigned to the requested address 192.168.0.110 and are able to deliver the required content.
- The LoadMaster uses the load balancing method configured, for example weighted round robin, to send the request on to one of the three Real Servers.
- Other items to note regarding the two-armed configuration are:
- Both eth0 (net side) and eth1 (farm side) interfaces are used. Additional ports go to the farm side for multi-armed configurations
- Implies that the LoadMaster (eth0) and server farm(s) are on separate logical networks, sometimes referred to as a NAT-based topology
- The server farm(s) may make use of non-routable (RFC1918) IP addresses
- Server NAT may be useful in such a configuration
- IP address transparency will function properly if clients are located on the same logical network as the LoadMaster in both NAT (common) and DSR (uncommon) configurations.
- Virtual Services may be created on any interface/VLAN that has an IP address configured.

Leveraging one port and configuring the “Additional Subnet” feature qualifies as two-armed.

High Availability (HA) Configuration

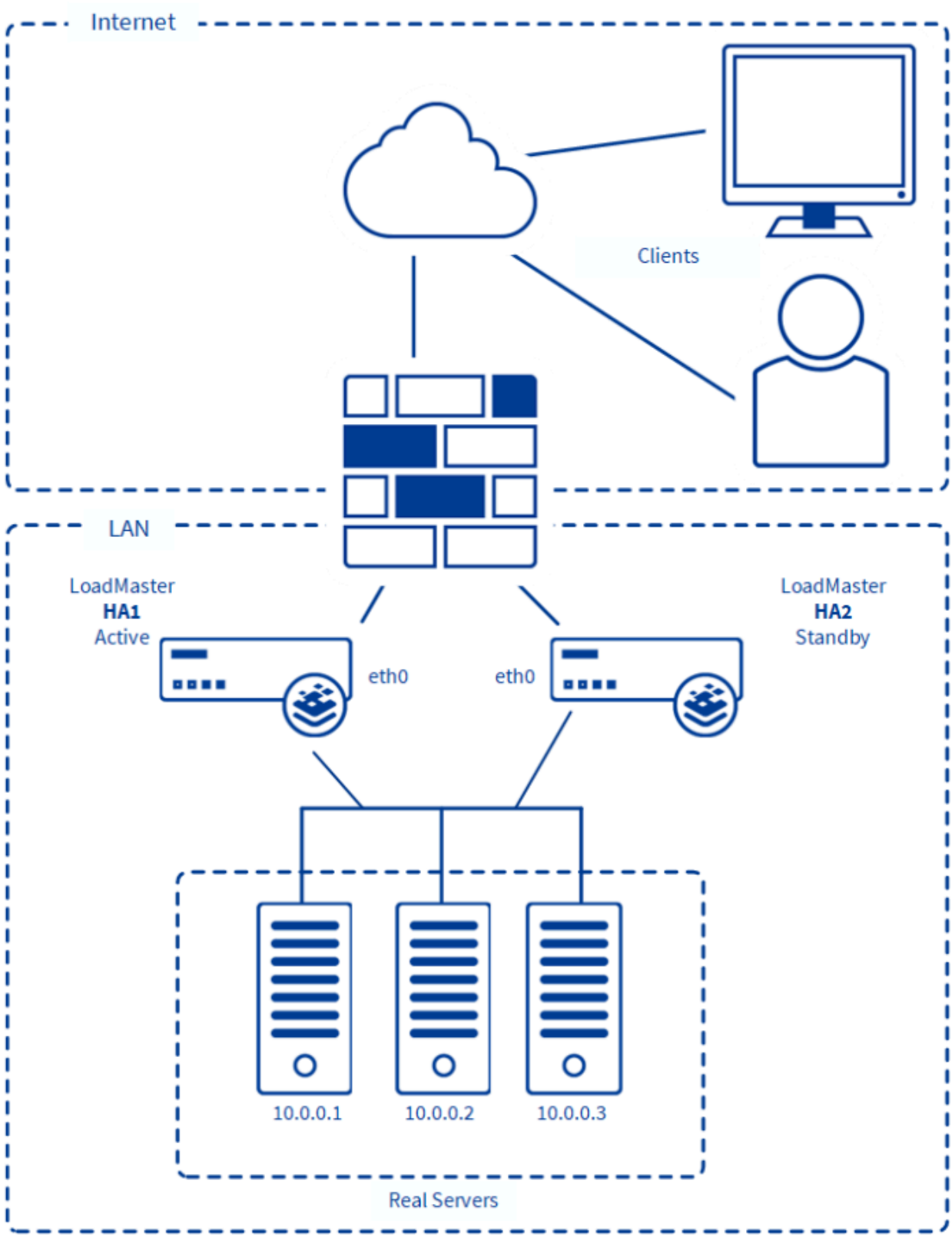
The High Availability feature of the LoadMaster guarantees the availability of your server farm. HA is achieved by a hot-standby, failover mechanism. Two identical LoadMaster units are integrated into the network as a HA pair. One machine serves as the active LoadMaster and the second one remains in a standby, idle state, always prepared to take over the activities from the active server. This HA pair appears as a single logical unit to the DMZ side and to the server farm side connections.

With a HA pair, each network interface has an individual IP address and one shared IP address which is shared with the partner unit. The shared IP address is identical for both LoadMaster appliances, though it is associated with only the active LoadMaster at any given time.

Note: If the LoadMaster is set as the default gateway of the servers, remember to use the shared address of the HA pair since this address will always be available.

During normal operation each node periodically sends health check messages over any interface that has the **Use for HA checks** option enabled to verify the availability of the peer appliance. Health checks are enabled unless they have been explicitly disabled. In the event that the active LoadMaster should fail, the standby appliance will become active and take over the task of balancing.

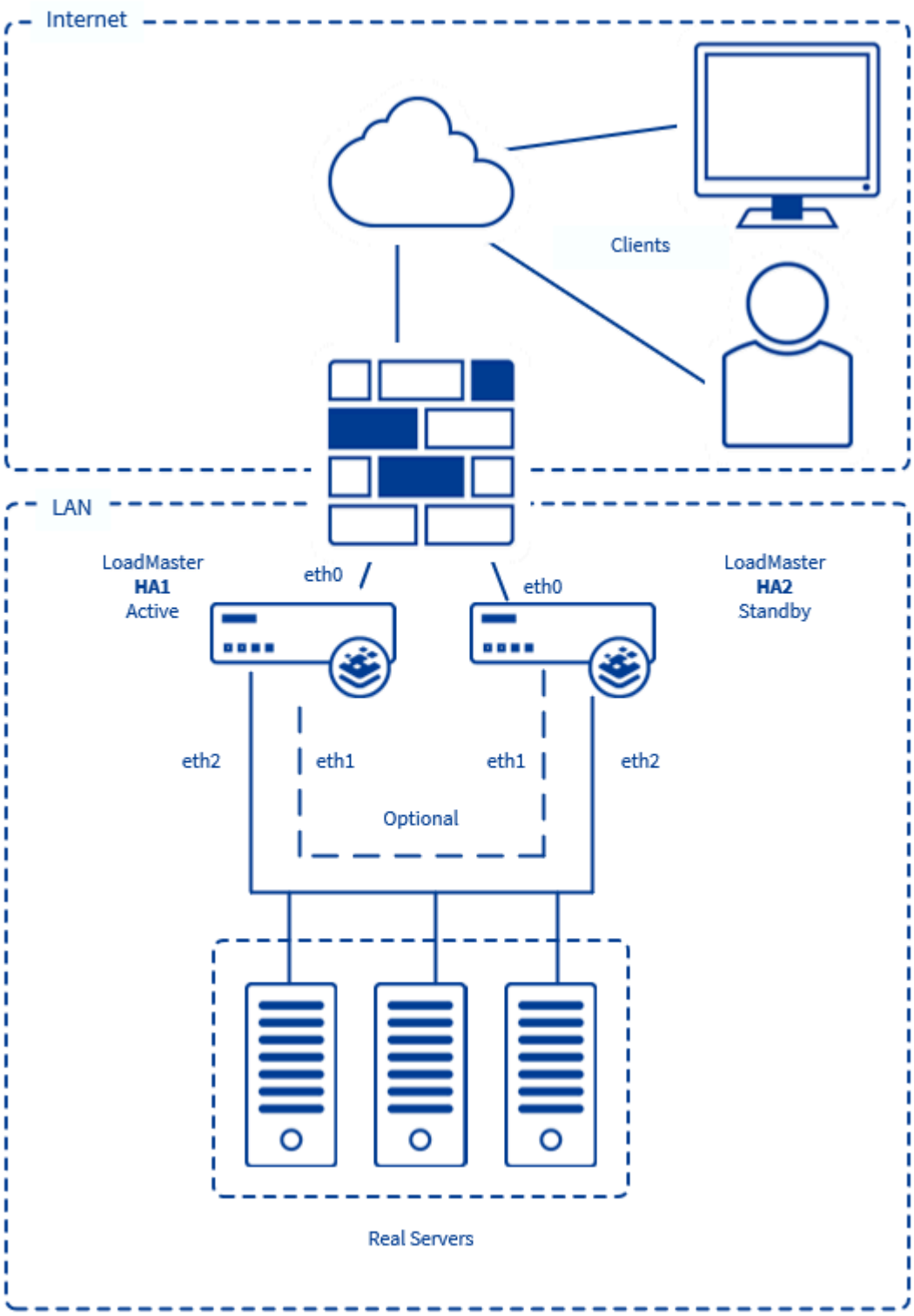
The topology for HA single arm looks like this:



LoadMasters HA1 and HA2 use eth0 to connect to the network (firewall) and to the servers and have one shared IP address between the two ports. Optionally, eth1 on each unit is directly connected using a patch

cable - the port is auto-sensing so it makes no difference if the cable is straight or reversed - and is used exclusively for additional HA health checking.

The topology for HA dual arm looks like this:



Both HA1 and HA2 use eth0 to connect to the network (firewall) and eth2 for connection to the servers. The two eth0 ports have one shared IP address and the two eth2 ports have a different shared IP address. Health checking between the two LoadMasters occurs between both eth ports. Optionally, eth1 on each unit may be directly connected using a patch cable for added HA health checking though it is quite unnecessary since there is already 2 health check routes between the HA pair.

Note: HA1 and HA2 must be on the same subnet with the same default gateway and be located within the same physical site. They must not be separated by an intra-site link and must use the same gateway to return traffic.

Note: Multiprotocol Label Switching (MPLS) networks are not supported in a HA setup.

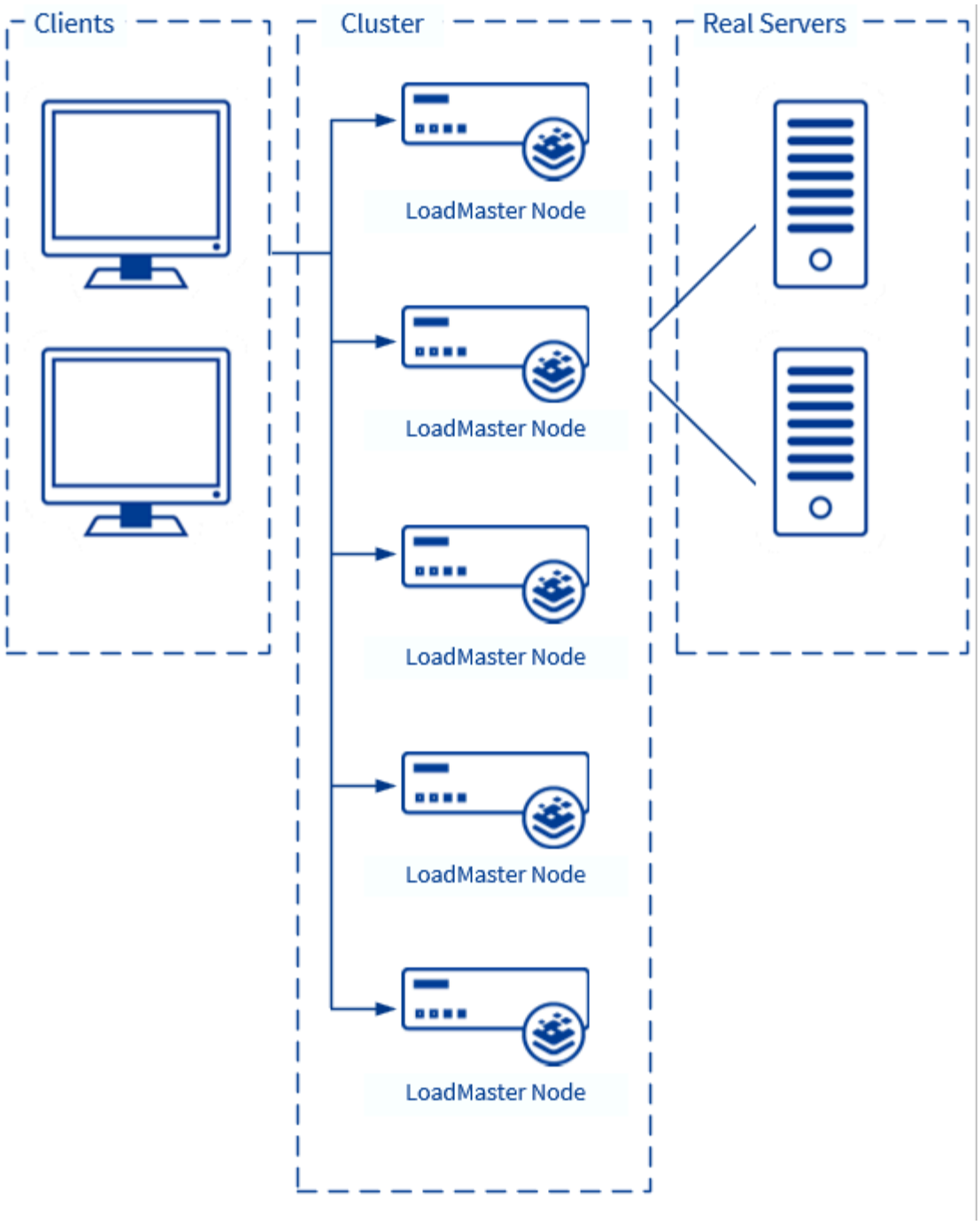
Running HA spanned across multiple subnets will not provide hardware redundancy in the event of a failure of the link between them. If traffic balancing between multiple sites is required, the GEO LoadMaster is the correct solution. This is Progress Kemp's DNS-based appliance that employs health checking to avoid site outages. For further details, refer to the GEO Feature Description.

Clustering

Clustering involves deployment of multiple LoadMaster instances (nodes) as a single management and control domain where each node is actively passing traffic. If any of these nodes fail - the remaining nodes continue to provide service to the remote client, providing resilience.

Clustering supports scaling - LoadMaster nodes can be added to or removed from a cluster so that capacity can be dynamically adjusted to match business requirements. Each additional LoadMaster node which is added to the cluster can provide a linear increase in throughput and performance. The maximum number of nodes that can be added to a cluster is defined by your license agreement.

An example use case is - if a website experiences higher than usual traffic during a holiday period and the amount of traffic cannot be predicted in advance - often the capacity is over-provisioned to handle these requests that come in spikes. One of the simplest solutions is to deploy LoadMasters in a cluster and then add new nodes as the capacity requirements (concurrent connections and performance requirements) increase.



With LoadMaster clustering, the load-balancing capability can be extended as needed by adding additional LoadMasters for the same Virtual Service. Traffic which is intended for a failed node is either passed on to an existing node or load-balanced across the remaining nodes.

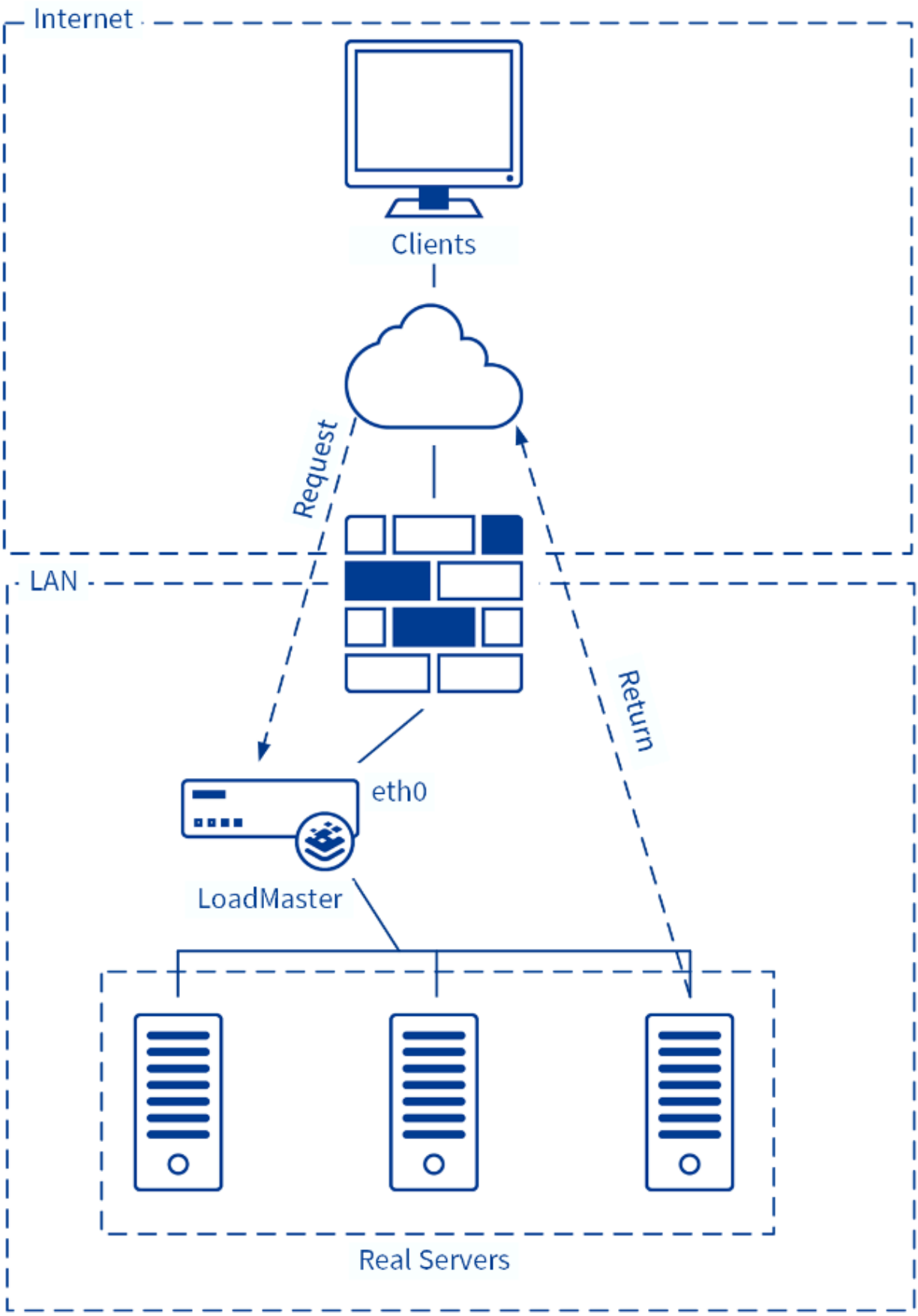
The LoadMasters all work in parallel. If a LoadMaster node fails, traffic is redistributed amongst the remaining nodes. Health checking is performed on each of the LoadMasters.

Most administrative changes should be made on the shared IP address interface. As soon as a change is made on the shared IP address, it is propagated to the other nodes. Local administration of the individual LoadMasters can also be performed by accessing the WUI of the units directly. The WUI of the local

LoadMasters contains limited configuration options.

For further information on clustering, refer to the LoadMaster Clustering, Feature Description.

Direct Server Return – DSR Configuration Example


Internet
Clients
Request
LAN
Return
eth0
LoadMaster
Real Servers

1. Incoming request intercepted by LoadMaster
2. Routed to Server 1
3. Response from Server 1
4. Response goes directly to Client without LoadMaster

This feature should be implemented only if the Real Servers need to respond to the clients directly, without going through the LoadMaster. In this configuration the Real Servers must have a path to the clients without going through the LoadMaster, for example an additional router in parallel with the LoadMaster.

Note: The only persistence option supported in a DSR configuration is Source IP. No Layer 7/Application features can be used with DSR. Also, DSR may be used only in a 1-arm configuration due to routing issues caused on the Real Server with the loopback interface in a 2-arm solution.

DSR uses a combination of MAT (MAC Address Translation) and a modified Real Server configuration. The RS is configured with an IP address as normal but it is also given the IP address of the VIP. Normally you cannot have two machines on a network with the same IP address. To get around this, the VIP address on a Real Servers must be configured so that the server does not respond to ARP requests on the VIP address.

For further information on how to configure Real Servers (both Linux and Windows) please refer to the Configuring Real Servers for DSR, Technical Note document.

3

Scheduling Methods

There are several load balancing methods provided by the LoadMaster, which are known as "Scheduling Methods" or “algorithms”. These are described in the sections below.

Round Robin

With this method, incoming requests are distributed sequentially across the server farm (cluster), that is, the available servers.

If this method is selected, all the servers assigned to a Virtual Service should have the similar resource capacity and host identical applications. Choose round robin if all servers have the same or similar performance and are running the same load. Subject to this precondition, the round robin system is a simple and effective method of distribution.

However, if the servers have different capacities, the use of the round robin system can mean that a less powerful server receives the next inquiry even though it has not yet been able to process the current one. This could cause a weaker server to become overloaded.

Weighted Round Robin

This method balances out the weakness of the simple round robin: incoming requests are distributed across the cluster in a sequential manner, while taking account of a static “weighting” that can be pre-assigned per server.

The administrator simply defines the capacities of the servers available by weighting the servers. The most efficient server, **A** for example, is given the weighting 100, whilst a much less powerful server (**B**) is weighted at 50. This means that Server **A** would always receive two consecutive requests before Server **B** receives its

first one, and so on.

Least Connection

Both round robin methods do not take into account that the system does not recognize how many connections are maintained over a given time. It could therefore happen that Server **B** is overloaded, although it receives fewer connections than Server **A**, because the users of this server maintain their connections longer. This means that the connections, and thus the load for the server, accumulate.

This potential problem can be avoided with the "least connections" method: requests are distributed on the basis of the connections that every server is currently maintaining. The server in the cluster with the least number of active connections automatically receives the next request. Basically, the same principle applies here as for the simple round robin: the servers related to a Virtual Service should ideally have the similar resource capacities.

Please note that in configurations with low traffic rates, the traffic will not balance out and the first server will be preferred. This is because if all the servers are equal, then the first server is preferred. Until the traffic reaches a level where the first server continually has active traffic, the first server will always be selected.

Least Connection Slow Start Time

For both the Least Connection and the Weighted Least Connection scheduling methods, when a Real Server is initially brought online, a period of time can be configured where the number of connections is initially restricted and gradually increased. This provides a ‘ramp-up time’ for a Real Server to ensure that it is not overloaded by a flood of connections upon startup.

This value is configured in the L7 configuration screen.

Weighted Least Connection

If the servers have different resource capacities the “weighted least connection” method is more applicable: The number of active connections combined with the various weights defined by the administrator generally provides a very balanced utilization of the servers, as it employs the advantages of both worlds.

This is, in general, a very fair distribution method, as it uses the ratio of the number of connections and the weight of a server. The server in the cluster with the lowest ratio automatically receives the next request.

Please note that the caveat for Least Connections regarding low traffic rates applies here as well.

Agent-Based Adaptive Balancing

The “resource-based (adaptive)” scheduling method periodically checks every Real Server in a Virtual Service for an integer value that describes the availability status of the Real Server in more detail than a health check alone. This allows the LoadMaster to apply more complex logic when making load balancing decisions.

For this method to work, each Real Server in a Virtual Service must populate a simple ASCII text file with an integer value that indicates the level of availability of the Real Server. For example, 0 indicates an idle server, while 100 indicates a fully loaded server. The LoadMaster retrieves this file on a regular interval from the Real Server using a HTTP GET request for the text file containing the integer.

One of the advantages of this method that makes it powerful is that the LoadMaster takes specific actions in response to specific return codes. For example:

- When the load reported from the Real Servers is above 0, the scheduling algorithm calculates a weighting ratio out of the collected load values and distributes the connections according to it. So, if excessive overloading of a server occurs, the weighting is readjusted transparently by the system. This is similar to the weighted round robin method, where a more even distribution is achieved by assigning different weights to the servers available.

- When the load values as reported by the servers are 0 (or the LoadMaster fails to GET an integer value from the server for whatever reason), the LoadMaster cannot build a representative sample load distribution to modify server weight. In this case, the LoadMaster switches temporarily to the round robin selection method. Once the Real Servers begin returning larger load values to build a representative traffic sample, the LoadMaster switches back to the adaptive method.

It is up to the administrator to configure the Real Server with some method of determining a load value and placing it into a text file for the LoadMaster to retrieve. Many customers use an 'adaptive agent script' that runs on the server specifically for this purpose.

For complete information on the integer protocol used to communicate Real Server availability status to the LoadMaster, in addition to sample adaptive agent scripts for both Linux and Windows systems, refer to the Writing a Resource Based Adaptive Server Agent Technical Note.

Resource-based (SDN Adaptive)

The LoadMaster contains adaptive load balancing technology which can be used with a Software Defined Networking (SDN) controller. In traditional networks, there is no end-to-end visibility of network paths and applications are not always routed optimally. The LoadMaster, integrated with an SDN Controller solution, solves this problem by making the critical flow pattern data available.

The LoadMaster pulls the Layer 2/Layer 3 information from the switches in the network using the Controller. The LoadMaster combines the Layer 2/3 information with the Layer 4/7 information to make more optimized traffic distribution decisions. The LoadMaster can be used to provide end-to-end visibility of network paths for optimal routing of applications across the server and switching infrastructure.

The Progress Kemp SDN solution provides greater efficiency by enabling:

- Application visibility to the network
- Network data to be pulled by the Application Delivery Controller (ADC)
- Adaptive load balancing

A Virtual Service which is using an adaptive scheduling method can be viewed as a control system. The intent is to achieve an evenly distributed load over the Real Servers and the controller calculates an error value from this (that describes the deviation from the desired even distribution). It also calculates a set of control values (Real Server weights) that are fed back into the system in a way to decrease the error value.

For more information on SDN adaptive load balancing, refer to the SDN Adaptive Load Balancing, Feature Description.

Fixed Weighted

The highest weight Real Server is only used when other Real Server(s) are given lower weight values. However, if the highest weighted server fails, the Real Server with the next highest priority number will be available to serve clients. The weight for each Real Server should be assigned based on the priority among the Real Server(s). If multiple Real Servers share the same highest weight, connections are forwarded to the first highest-weighted, available Real Server.

Weighted Response Time

The traffic is scheduled using the weighted round robin method. The weights used by the weighted round robin method are calculated using the response times from health check requests.

Each health check request is timed to see how long it takes to respond. Please note that is assumed that the speed of the health check depends on how slow the machine is which may not always be the case.

The total response times of all the Real Servers on the Virtual Service are added together, from which the weight of the individual Real Server is calculated.

The weights are recalculated approximately every fifteen seconds.

Source IP Hash

When using the **source IP hash** scheduling method with Layer 4 (L4) Virtual Services, LoadMaster uses a prepopulated hash table to determine how client traffic is distributed across Real Servers.

For L4 source IP hash scheduling:

- LoadMaster uses a fixed hash table containing 4096 destination slots.
- Each Real Server is assigned a number of slots based on its configured Server weight.
- The weights of all Real Servers associated with the Virtual Service are tallied together to determine how the 4096 slots are distributed.

This prepopulation directly affects how source IP addresses are mapped to Real Servers.

By default, each Real Server is assigned a weight of **1000**.

When multiple Real Servers are configured using the default weight, the total combined weight may not evenly divide into 4096.

In such cases, LoadMaster allocates slots sequentially based on Server weight until all 4096 slots are assigned. Any remaining slots are assigned starting again from the first Real Server.

As a result, some Real Servers particularly those added later may receive significantly fewer connections or no connections at all.

Example 1: Four Real Servers, weight 1000 each

- rs1: 1000 slots
- rs2: 1000 slots

- rs3: 1000 slots
- rs4: 1000 slots
- rs1: remaining 96 slots

Example 2: Five Real Servers, weight 1000 each

- rs1: 1000 slots
- rs2: 1000 slots
- rs3: 1000 slots
- rs4: 1000 slots
- rs5: 96 slots

In the second example, the fifth Real Server receives a very small share of the hash table, which can lead to minimal or no traffic being forwarded to that Server.

When using source IP hash scheduling, ensure that the sum of all Real Server weights is a factor of 4096.

Users should adjust Real Server weights rather than relying on the default value of 1000 based on the number of configured Servers. This ensures an even distribution of hash slots and prevents traffic imbalance.

4

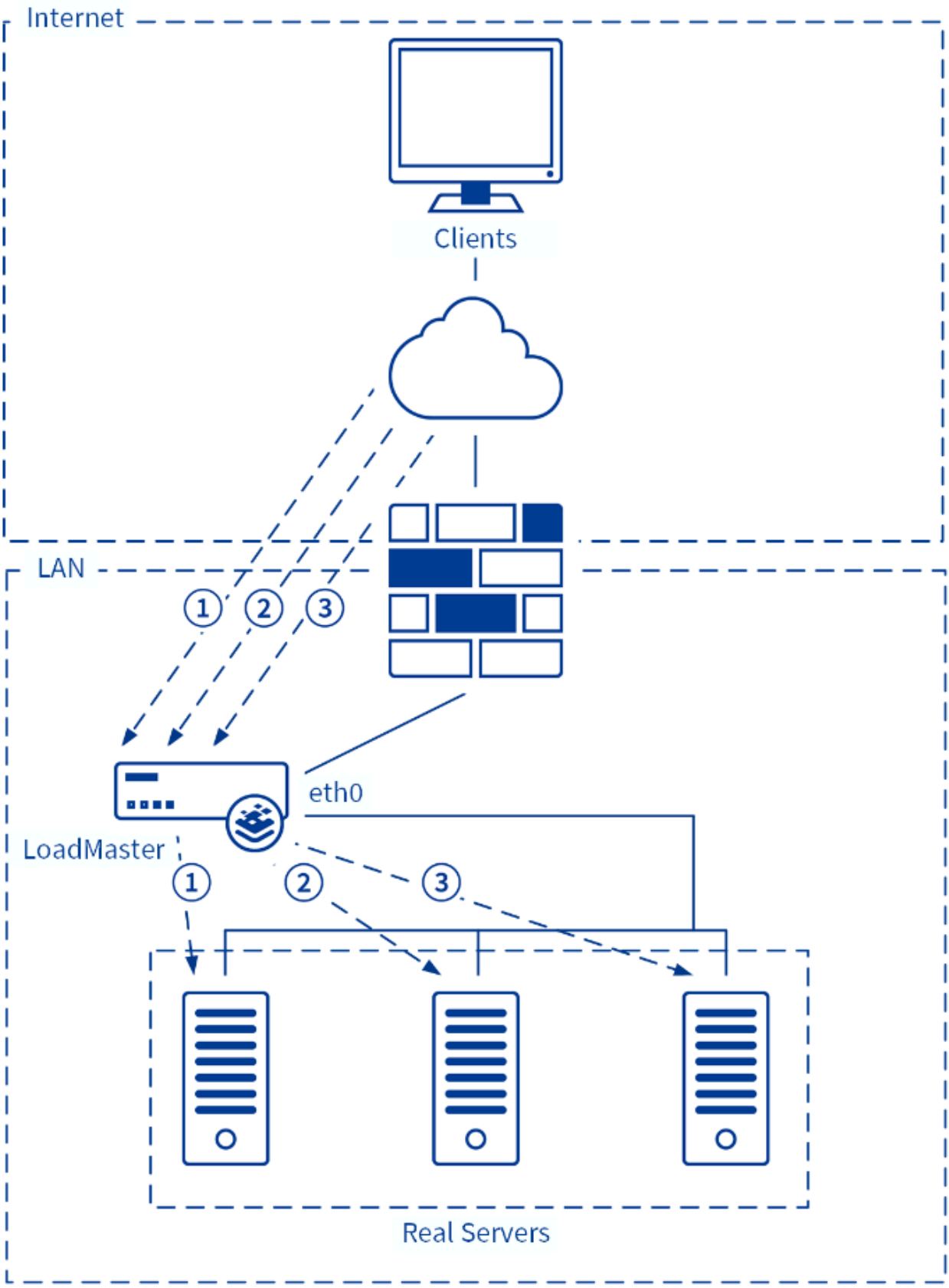
Persistence

Refer to the following sections for details on the LoadMaster persistence functionality.

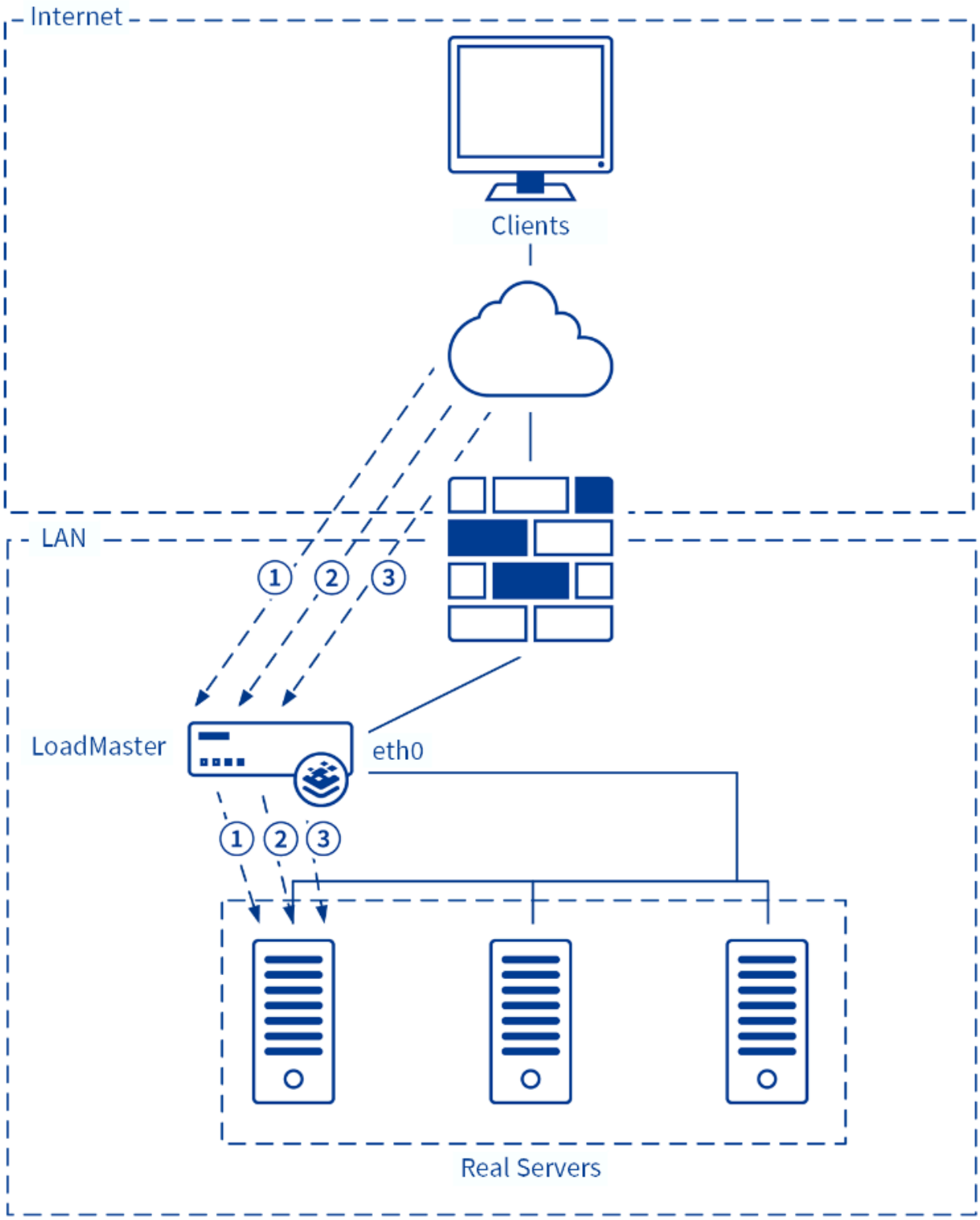
Introduction to Persistence

Persistence – which can also be referred to as “affinity”, “server affinity”, or “server sticky” -- is the property that enables all requests from an individual client to be sent to the same server in a server farm. Persistence is not turned on by default, but it is an option configurable for each Virtual Service.

Without persistence, the LoadMaster will direct traffic according to the load balancing algorithm, such as round-robin, weighted round-robin, etc. (diagram below).


Internet
Clients
LAN
1
2
3
eth0
LoadMaster
1
2
3
Real Servers

With persistence, the LoadMaster will direct new connections according to the load balancing algorithm, but returning connections will go to the same server (diagram below).



Contents

- How Do I Know If I Need Persistence?
- Timeout
- Layer 7 Persistence Methods
- Persistence and HTTPS/SSL
- Port Following

How Do I Know If I Need Persistence?

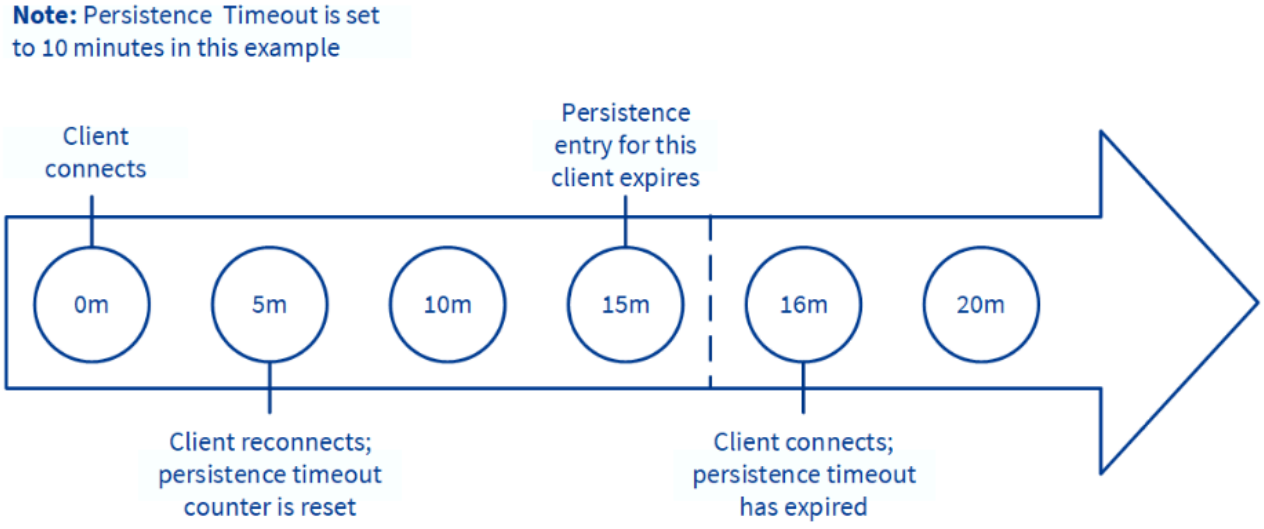
If the site you have is an interactive site, then you will probably need persistence. This is especially true for sites that require some type of login. If the site you are running is static, serving only static text and images, then you may not need persistence. In most cases, persistence cannot hurt even if you do not need it.

The session handling mechanism for many website programming languages (ASP, PHP, and so on) are known as “stateful”. There is a unique session established for the user, and that “state” is kept on the same server. This stateful information, which can include everything from login credentials to the content of a shopping cart, is typically not shared among servers. So, when using multiple servers it is important to keep an individual user tied to a specific web server for the duration of the interaction, and that is where persistence comes in.

Timeout

For each persistence method, there is a configurable timeout value that determines how long the persistence for each user is honored, selectable from one minute to 28 days.

This timeout clock is started when the initial connection is established. The persistence timeout value is updated if the client reconnects within the timeout period. For example, if the persistence timeout is set to 1 hour and the client starts a connection at 2pm, if the client disconnects and then reconnects before 3pm they will still persist to the same Real Server. Also, the persistence record is updated to reflect this and the persistence countdown timer is reset back to 1 hour for this client.



If a client made connections to the Virtual Service repeatedly within the timeout period, the persistence would be honored indefinitely. For instance, given the following scenario:

- Persistence Timeout is set to 10 minutes
- A user makes several requests in the course of 20 minutes, but the time between connections is always less than 1 minute

The request should be sent to the correct Real Server, as long as it is available (that is, passing health checks).

If the user goes idle for 20 minutes, then the next connection will be counted as a new session, and may be sent to a different server, depending on scheduling. If the connection is opened for more than 10 minutes and the client disconnects and reconnects, the persistence record would have expired, the LoadMaster will create a new persistence entry for that client and possibly send the client to a new Real Server. This is due to the fact that the persistence countdown starts once a connection is established, not at the closing of the connection.

If you are experiencing persistence issues, this may be due to the fact that the persistence timeout is not long enough. If this is not long enough, then the timeout value should be set for a higher amount. In general, matching this value to your server timeout value is recommended.

Layer 7 Persistence Methods

These are methods that look beyond the IP address and port and provide a range of options to achieve layer 7 persistence.

Server Cookie Persistence

The Server Cookie option is a Layer 7 feature that uses existing cookies generated from the server to determine which server to send users to. This method is sometimes referred to as “passive cookie”, as the LoadMaster does not generate or manage the cookie, it only observes the cookie in the HTTP stream.

With Server Cookie persistence, you will need to configure the Cookie Name option so the LoadMaster knows which cookie to refer to. For Server Cookie persistence to work best, the cookie generated by the server should have a unique value for each individual user.

Active Cookie Persistence

The Active Cookie method is a Layer 7 feature that uses cookies like the previous method, but with Active Cookie the cookies are generated by the LoadMaster, not the server.

When a connection comes into a LoadMaster Virtual Service configured with Active Cookie, the LoadMaster looks for a specific cookie. If that cookie is not there, the LoadMaster inserts it into the HTTP stream with a Set-Cookie directive. Existing cookies are not affected.

As with the Server Cookie persistence method, the value for the LoadMaster-generated cookie is unique to each user, allowing the LoadMaster to differentiate between users.

A benefit of this method is that no cookies need to be managed or generated by the servers, relieving the

burden of server configuration. To gain better dispersion per client connection you can enable the “Add Port to Active Cookie” feature in the L7 configuration.

With Active Cookie persistence, the cookie is valid for the session or until the persistence time expires. For example, if using Active Cookie persistence with the persistence timeout set to 10 minutes and the client connects at 2pm, then disconnects and reconnects at 2.05pm – this would reset the persistence timeout value. If the client tries to connect to a Virtual Service after the persistence timeout has expired, they would present the old cookie. The LoadMaster will check its persistence table and see that it does not have a valid entry. The LoadMaster would then generate a new cookie for the client and would update its persistence table.

Server Cookie or Source IP Persistence

The Server Cookie or Source IP setting is identical to the Server Cookie setting, but with the additional fall-back method of source IP address. If, for any reason, the expected cookies are not present (this can happen when a client browser is configured to refuse cookies), then the source IP address will be used to determine persistence.

Active Cookie or Source IP Persistence

The Active Cookie or Source IP setting is identical to the Active Cookie Persistence. If, for any reason, the expected cookies are not present, then the source IP address will be used to determine persistence.

All things being equal, if you are going to use Layer 7 persistence, this is the recommended method. It requires no configuration on the servers, the LoadMaster manages all persistence-related cookies, and it falls back onto source IP address in cases where cookies are rejected by the client.

Hash All Cookies Persistence

The Hash All Cookies method creates a hash of the values of all cookies in the HTTP stream. Cookies with the same value will be sent to the same server for each request. If the values change, the connection will be treated as a new connection. The client will then be allocated to a server according to the load balancing algorithm.

Hash All Cookies or Source IP Persistence

Hash All Cookies or Source IP is identical to Hash All Cookies, with the additional feature that it will fall back to Source IP persistence in the event that no cookies are in the HTTP string.

Source IP Address Persistence

Source IP Address persistence uses the source IP address of the incoming request to differentiate between users. This is the simplest method of persistence, and works for all TCP protocols, including those that are not HTTP related.

Weakness of Source IP Address

There are situations where Source IP persistence may be undesirable or even ineffective in properly keeping

persistence. These situations include:

- When many (or all) users appear to come from a single IP address
- When a user switches IP addresses

The first case is often encountered when a significant number of user requests traverse a single proxy, and thus appear to come from a single IP. With Source IP persistence, this would mean that all of those users would appear as a single user.

Another way this might occur is when all of the client requests come over the Internet from a single office. Office routers typically NAT all office systems to one IP address, so again, all users and all requests would appear to be a single user. This can result in uneven load balancing, since new user sessions arriving would all be directed to the same Real Server, without being balanced.

The second case is a largely historical concern, having to do with proxy servers at some of the mega-Internet Service Providers (ISPs), for example AOL and Earthlink. In some cases, proxy configuration or any number of networking issues, might switch IP addresses from time to time. When the IP address changes, the user appears as a different user to SRC persistence.

In each of these cases, Layer 7 persistence would solve the issue, regardless of what IP they came from. However, this only works for the HTTP protocol (and HTTPS/SSL when the session is terminated at the LoadMaster).

Super HTTP

Super HTTP persistence functions by creating a unique fingerprint of the client browser and uses that fingerprint to preserve connectivity to the correct Real Server. The fingerprint is based on the combined values of the User-Agent field and, if present, the Authorization header. Connections with the same header combination will be sent back to the same Real Server.

URL Hash

With URL Hash persistence, the LoadMaster will send requests with the same URL to the same server.

HTTP Host Header

With HTTP Host Header persistence, the LoadMaster will send all requests that contain the same value in the HTTP Host: header to the same server.

Hash of HTTP Query Item

This method operates that the named item being inspected is a Query Item in the Query String of the URL. All queries with the same Query Item value will be sent to the same server.

Selected Header

With Selected Header persistence, the LoadMaster will send all requests that contain the same value in the specified header (for example, the X-Forwarded-For header - if it was added before it reaches the

LoadMaster) to the same server.

SSL Session ID

SSL Session ID is a persistence method that may be used with SSL services even if they are not offloaded. This relies on the client maintaining the same SSL Session ID for a complete user session.

Note: Browser support of this is spotty. As a result, this is not recommended for use with HTTPS services. In order to utilize this persistence method, the Service Type must be set to Generic.

Note: **SSL Session ID** will only be available as a persistence mode if the Virtual Service has a **Service Type** of **Generic** and **SSL Acceleration** is disabled.

UDP Session Initiation Protocol (SIP)

This persistence mode is only available in a UDP Virtual Service when **Force L4** is disabled. SIP uses request and response transactions, similar to HTTP. An initial INVITE request is sent, which contains a number of header fields. These header fields can be used for persistence. When this mode is selected in the LoadMaster, a text box called **Header field name** will appear. The header field that is to be used as the basis for the persistence information should be entered here.

Persistence and HTTPS/SSL

With HTTPS/SSL, there are a few things to consider. If you are not terminating the SSL session at the LoadMaster, then your only options are Source IP Address persistence or SSL Session ID Persistence. Since the stream is encrypted in a non-terminated session, the LoadMaster cannot look at the HTTP headers or other Layer 7 information.

If you are terminating the HTTPS/SSL session at the LoadMaster, then any of the LoadMaster persistence options can be used. Since the HTTPS/SSL session is terminated, the LoadMaster sees all of the unencrypted traffic, and is able to look at the HTTP stream. This is true even when you are terminating the HTTPS/SSL session at the LoadMaster and then re-establishing an SSL session with the Real Servers.

Port Following

When using “shopping cart”-like services where a user selects items and adds them to a list, any of the previous types of persistency can be used. When the user then decides to pay for the items, this is normally performed using a secure SSL (https) service. When port following is turned on, the Real Server where the “shopping cart” connection is active will be selected for the SSL session. This selection will only occur when a connection is still open from the same client (as determined by the source IP address), and if the SSL service has the same IP address as the “shopping cart” service.

For example, if a connection is made to the HTTP service of www.somewebsite.com, and then a new SSL connection is made to the same address, then the SSL session will be directed to the same Real Server as the original HTTP service.

Port following can be done between UDP and TCP connections if desired.

5

Application Front End

Application Front End is a group of features that revolve around web application delivery and network optimization. The introduction of the LoadMaster Application Front-End Services (AFE) solves very core requirements by providing better bandwidth and server utilization while allowing LoadMaster to remain a transparent load-balancing appliance that is easy to deploy and manage. LoadMaster AFE Services include:

- Intrusion Prevention System (IPS)
- Caching
- Data Compression

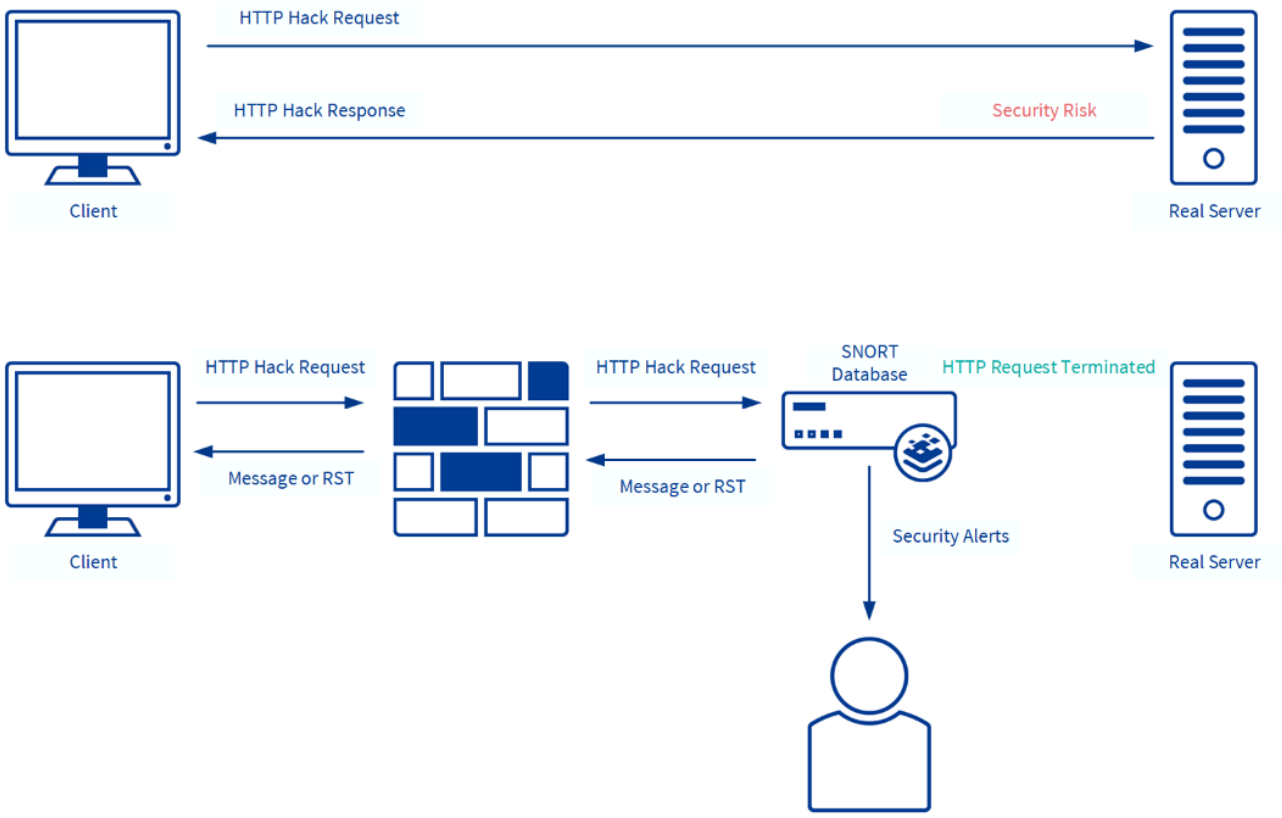
Each feature can be deployed per web Virtual Service.

Note: AFE features are license–based. If you do not have these features and you need them, please contact a Progress Kemp Sales Representative.

Contents

- Intrusion Prevention System
- Caching
- Data Compression

Intrusion Prevention System



The LoadMaster is an established, hardened Internet appliance with HTTP intrusion prevention. In addition to Secure Socket Layer (SSL), Denial of Service support offered by the LoadMaster, the Intrusion Prevention System (IPS) service will provide in-line protection of Real Server(s) by providing real-time mitigation of attacks and isolation of Real Server(s). Intrusion prevention is based on the industry-standard SNORT database and provides real-time intrusion alerting.

Note: The LoadMaster supports SNORT rules version 2.9 and below.

IPS is available for HTTP, offloaded HTTPS, and re-encrypted HTTPS Virtual Services.

Contents

- Intrusion Handling
- Detection level
- Warnings
- Intrusion Alerts
- Updating IPS Rules

Intrusion Handling

There are two options for handling of requests that match a SNORT rule: Drop Connection or Send Reject. Both options prevent the request from reaching the Real Server(s); this option configures the response returned to the client sending the malicious request.

Drop Connection Intrusion Handling

A rule match will generate no HTTP response. The TCP connection will terminate - no HTML content will be delivered to the client.

Send Reject Intrusion Handling

Once a rule is matched the response to the client will be set to HTTP 400 “Invalid Request” and the corresponding exploit note will be delivered to the client in a HTML document.

Sample Request: http://<VIP>/modules/articles/index.php?cat_id=SQL

Sample Response: <html><head><title>400 Invalid Request</title></head><body>Invalid Request: COMMUNITY WEB-PHP Xoops module Articles SQL Injection Exploit</body>

Detection level

The aggressiveness of rule matching can be configured globally for the appliance as per SNORT priority level.

- **Low** = Only logging with no rejection
- **Default** = Priority 1 (high) rules are block all else is logged
- **High** = Priority 1 (high) and 2 (medium) rules are block all else is logged
- **Paranoid** = All priority levels are blocked and logged

Warnings

The IPS system will throw out any malicious connections, but there are some requests that are not exactly dangerous, but an indication that something may be wrong. These are not blocked and by default, these are not logged, turning on the WARNING option will allow the logging of these requests.

Examples for non-dangerous operations are requests that are specified as misc-activity in the snort rule file:

Uri: "/OvCgi/OpenView5.exe?Context=Snmp&Action=Snmp&Host=&Oid="

which is described as "WEB-MISC HP OpenView Manager DOS" and is only suspicious.

Intrusion Alerts

All intrusion alerts are recorded in the system and warning logs. Alert notification can also be obtained by

syslog facility, the minimum level is Notice Host, and email alert facility, the minimum level is Notice Recipient. It is recommended that critical system messages like intrusion alerts be recorded by a syslog facility for records retention.

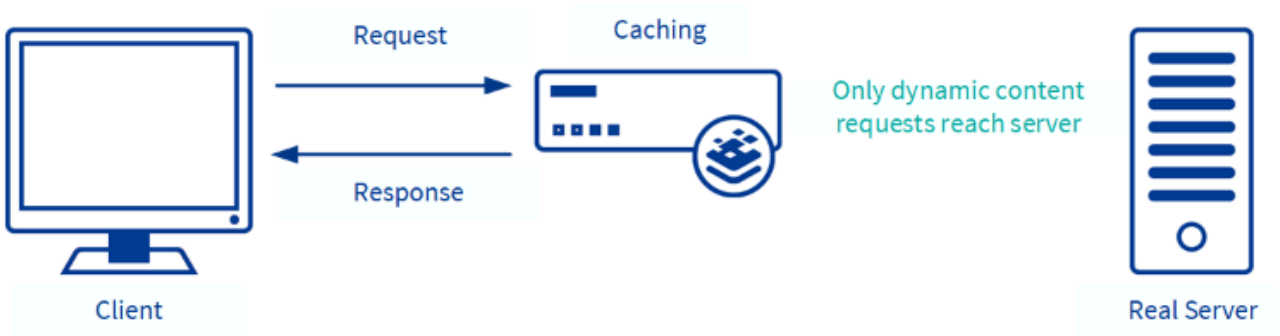
Updating IPS Rules

Rules can be downloaded from www.snort.org. Once a new rule set has been obtained or created you can load the rule set by using the WUI and navigating to **Certificates & Security > IPS / IDS**. Using the **Choose File** button, select the downloaded community-rules file and click **Install new Rules**. The community-rules files should be encoded in a Tar and Gzip file end with the tar.gz extension and containing a directory named "community-rules", LoadMaster will un-compress this file and reload the new rule files (tar.gz is the standard format for rules download from www.snort.org). Installing a new rules file will replace the current rules. LoadMaster ships with the Community Rules under General Public License (GPL) by default.

Caching

By deploying LoadMaster caching, your organization can greatly reduce web traffic to the Real Server(s), saving on bandwidth in-front of your Real Server(s).





Caching is available for HTTP and offloaded HTTPS Virtual Services.

Note: HTTP/HTTPS requests with no-cache headers will bypass the cache, following RFC 2616. Cache is

filled in a delayed manner - please allow a few seconds for static content to be cached.

Note: In accordance with RFC 2616, URLs which contain query strings (those containing a question mark symbol (**?**) in the rel_path part) will not be cached.

Cache is only stored in memory - it is not stored on disk.

Flushing Cache

LoadMaster will not monitor file changes on the Real Server and auto-reload the cache maintained within the Virtual Service. You can force reload the cache by deselecting and selecting the **Enable Caching** checkbox. You can also reload a cached object, sending a non-cache request. Most browsers support this by holding the left **Shift** key and clicking reload (or pressing **F5**).

Maximum Cache Size

The amount of global memory (in MB) available for caching can be configured; values have a linear relation to actual memory. To configure this, navigate to **Virtual Services > Cache Configuration**.

The default value for the **Maximum Cache Size** field is **100** MB.

The maximum value for the **Maximum Cache Size** field is 20% of the total memory allocated to the LoadMaster. For example, if the LoadMaster has 4 GB (4,096 MB) of memory, the maximum value of the **Maximum Cache Size** field is **819** MB.

To check the total memory of the LoadMaster, go to **Statistics > Real Time Statistics**.

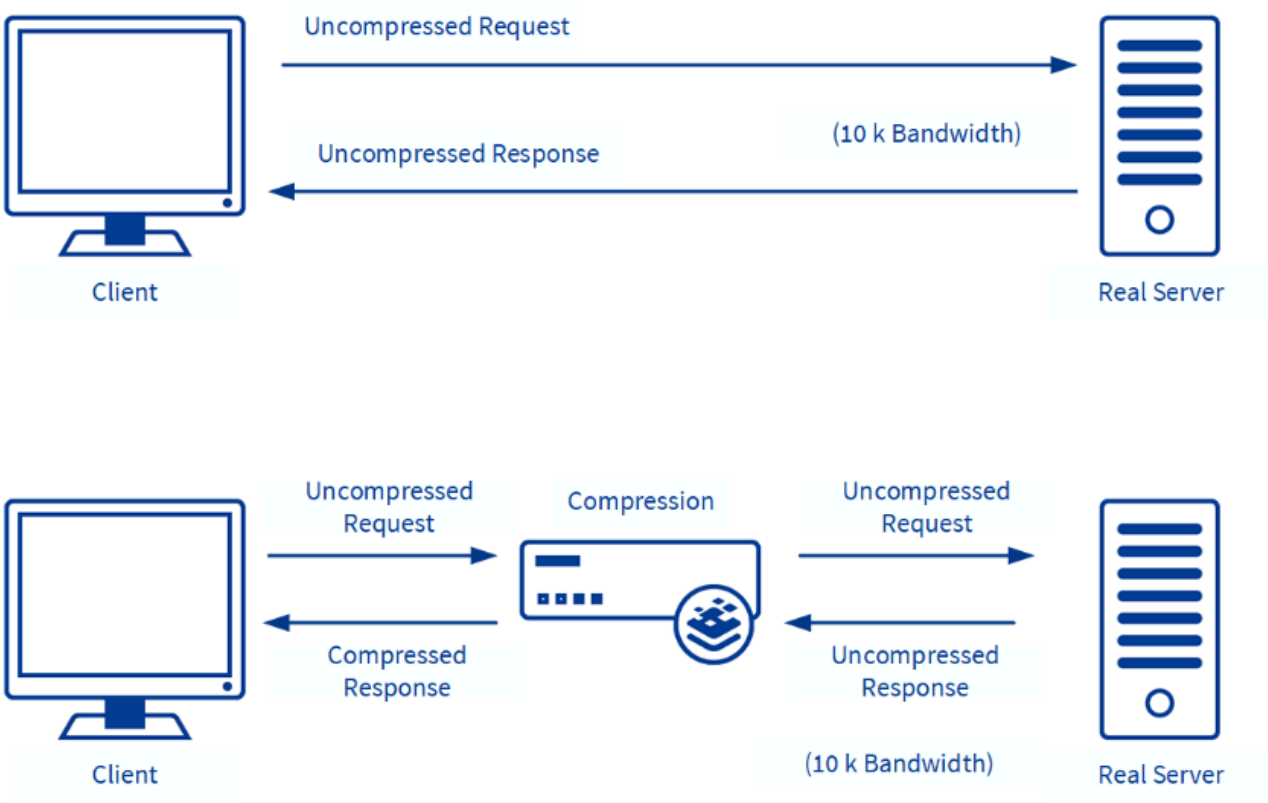
Data Compression

The LoadMaster data compression feature reduces the amount of data to be transferred for HTTP objects by utilizing gzip compression which is available in all modern web browsers. Leveraging Lempel-Ziv (LZ) compression and HTTP/1.1 GNU zip (gzip) content encoding reduces bandwidth utilization for high compression files such as text files (HTML, CSS, and JavaScript).

Data compression allows the LoadMaster to compress the application payload per request, reducing network bandwidth consumption without degrading content quality and response time resulting in an improvement for the end-users' overall experience. Data compression is supported on all files. Compression ratios vary by file type.

Note: Compression is not recommended for files 100Mb or greater in size.

Note: Compression only happens once - if the server already compressed the response, the LoadMaster will not compress it again.



Note: The compression feature should be deployed simultaneously with the caching feature to reduce the real-time inline compression requirements. Using only compression can potentially bottleneck the Virtual Service throughput depending on the hardware platform.

Compression can be enabled per HTTP, offloaded HTTPS, and re-encrypted HTTPS Virtual Services.

Compression depends on clients having gzip support. You can verify that a compressed connection to the Virtual Service exists by tracing the client HTTP traffic. If you can locate the **Content-encoding:gzip** header from the LoadMaster your client communication to the LoadMaster is compressed.

6

SSL Acceleration/Offloading

The LoadMaster series offers SSL termination/acceleration for Virtual Services. With SSL acceleration, the SSL session is terminated at the LoadMaster.

The LoadMaster Supports SSL 3.0, TLS 1.0, TLS 1.2, and TLS 1.3.

There are two primary benefits to SSL acceleration:

- The LoadMaster offloads the SSL workload off the Real Servers
- The LoadMaster performs Layer 7 processing: persistence or content switching

Without terminating the SSL session at the LoadMaster, the headers and content cannot be read so persistence cannot be done. The only consistently reliable persistence method available when the SSL session is not terminated at the LoadMaster, is Source IP.

With SSL acceleration, the LoadMaster uses a specialized processor to perform the SSL functions. With this SSL acceleration hardware, the LoadMaster handles SSL connections as easily as it handles non-SSL connections.

All LoadMasters can perform SSL termination. There are two types of SSL termination capabilities:

- Hardware SSL
- Software SSL

Note: Only software SSL is available in Virtual LoadMasters.

Functionally, hardware and software SSL are the same. The difference is - the part of the LoadMaster that

handles the actual cryptographic functions associated with SSL operations.

With software SSL, the LoadMaster's general processor handles encryption/decryption tasks. These tasks are shared with other tasks that the LoadMaster performs, such as load balancing, health checking, and other administrative tasks. Because SSL operations are CPU-intensive, software SSL is sufficient for low levels of SSL traffic but insufficient for higher levels of SSL traffic. Higher connection rates of SSL on a software SSL LoadMaster may degrade overall performance of the LoadMaster.

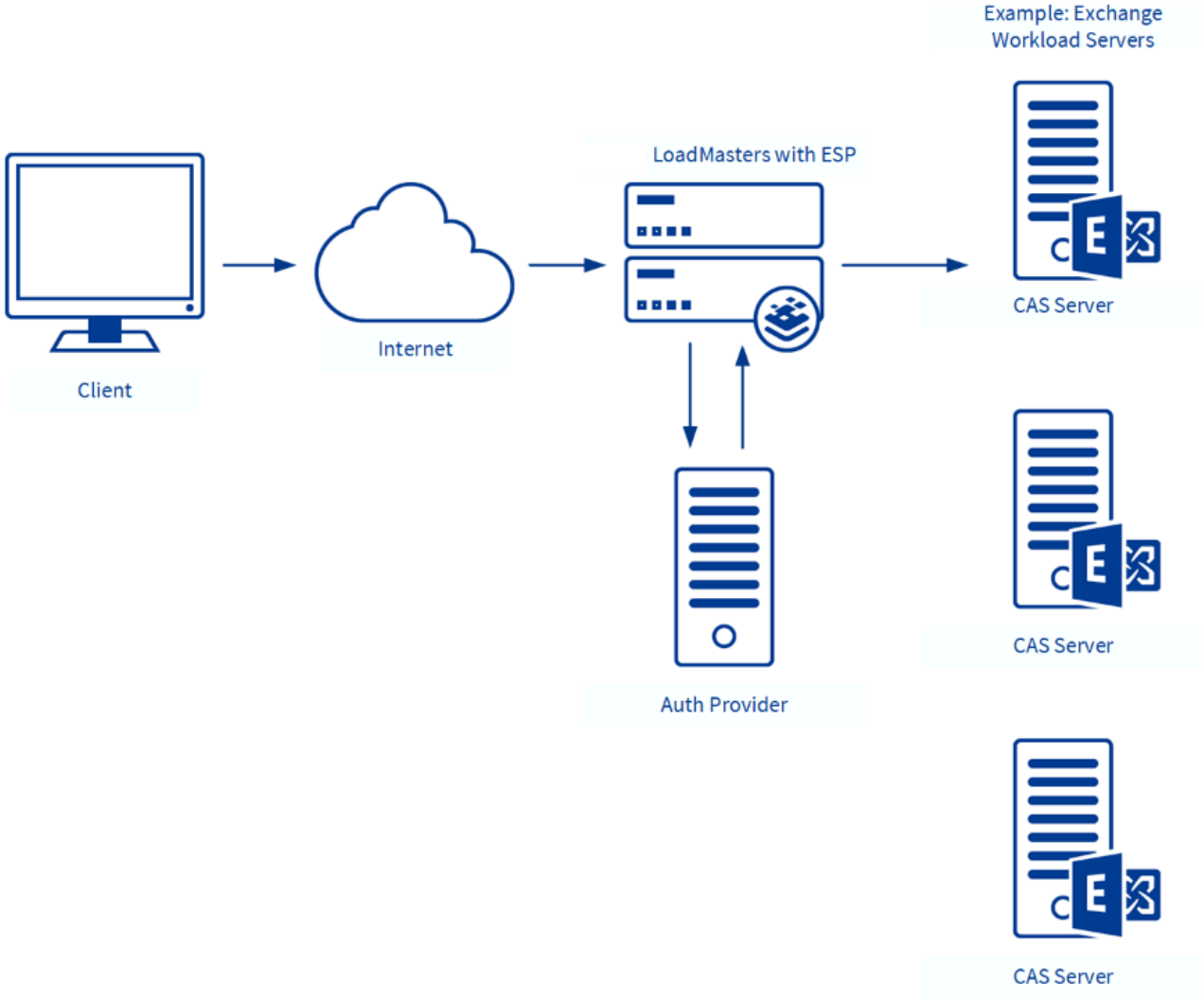
With hardware SSL, the LoadMaster has a separate specialized processor which handles all SSL functions. No matter what level of SSL connections, the LoadMaster's general processor is not burdened. This specialized hardware is purpose-built for SSL, and can handle extremely high connection rates (TPS) of SSL traffic.

For more information on SSL, refer to the SSL Accelerated Services, Feature Description.

7

Edge Security Pack (ESP)

The Edge Security Pack (ESP) pack delivers a solution using the LoadMaster line of load balancers to customers who would have previously deployed Microsoft's Threat Management Gateway (TMG) to publish their Microsoft applications.



ESP offers the following key features:

- End point authentication for pre-auth
- Persistent logging and reporting for user logging
- Single Sign On (SSO) across Virtual Services
- LDAP authentication from the LoadMaster to the Active Directory
- Basic and form-based authentication communication from a client to the LoadMaster
- Remote Access Dial In User Service (RADIUS) authentication
- RSA SecurID two-factor authentication
- Kerberos Constrained Delegation (KCD) authentication
- Client certificate authentication
- Dual-factor authentication

End Point Authentication for Pre-Auth

Clients who are trying to access Virtual Services on the LoadMaster will have to provide authentication information which will be used by the ESP to validate the clients' right to access the service. In the event of success, the client is enabled to access the service. In the event of failure, the client will be blocked until valid credentials are provided.

Persistent Logging and Reporting for User Logging

When clients try to access a service this will be logged on the LoadMaster as part of the ESP. This allows monitoring by the administrator.

Single Sign On Across Virtual Services

The LoadMaster is designed to handle multiple virtual services supporting unique workloads. These Virtual Services can be joined together by associating them with the same Single Sign-On (SSO) Domain.

Note: The Virtual Services need to be on the same domain for this to work, for example **ecp.example.com** and **www.example.com**.

SSO in ESP will enable clients to only enter the authentication information when accessing the first Virtual Service and then this same information will be used to access other services associated with the Single Sign-On Domain. Therefore, a client accessing Exchange will also be able to access SharePoint and other workloads if they are associated with the same Single Sign-On Domain.

LDAP Authentication from the LoadMaster to Active Directory

Active Directory is the standard Authentication Provider for Microsoft workloads. LoadMaster supports the key connection types between the LoadMaster and Active Directory.

Basic Authentication Communication from a Client to the LoadMaster

LoadMaster with ESP currently supports basic and form-based authentication between the client and the LoadMaster, providing clients with an optimum authentication experience. In future releases, there are plans to also support NTLM.

Large and small businesses are deploying large numbers of internet-facing applications to support ever expanding business requirements. This rapidly growing number of servers needs to be scalable and highly reliable. Above all, the access to these servers and services needs to be secure. With the addition of the ESP, the LoadMaster will continue to deliver on customer security requirements for internet-facing applications in a world without TMG, while continuing to address requirements for feature-rich and cost-effective scalability and high reliability.

RADIUS Authentication

A Remote Access Dial In User Service (RADIUS) server can be used to authenticate users who log in to the LoadMaster. The LoadMaster passes the user's details to the RADIUS server and the RADIUS server informs the LoadMaster whether the user is authenticated or not.

RADIUS in Windows Server 2008 R2 is done with network policy and access services.

For more information, refer to the RADIUS Authentication and Authorization, Technical Note.

RSA SecurID Two-Factor Authentication

As part of the Edge Security Pack (ESP), the LoadMaster supports the RSA SecurID authentication scheme. This scheme authenticates the user on an RSA SecurID Server. When RSA is enabled as the authentication method, during the login process the user is prompted to enter a password that is a combination of two numbers – a Personal Identification Number (PIN) and a token code which is the number displayed on the RSA SecurID authenticator (dongle).

There are two additional challenge-response modes: next token and new PIN.

For more information, refer to the RSA Two Factor Authentication, Feature Description.

Kerberos Constrained Delegation (KCD) Authentication

When using KCD as the authentication protocol, the LoadMaster provides seamless access to protected resources in a Kerberos realm even when credentials provided are not directly valid for such an environment.

The KCD authentication protocol is used to confirm the identity of the users that are attempting to access resources on a network. KCD authentication uses tickets that are encrypted and decrypted by secret keys and do not contain user passwords. These tickets are requested and delivered in Kerberos messages. When the user's password is not provided, a trusted administrator user account is used to get tickets on behalf of services and users.

For more information refer to the Kerberos Constrained Delegation (KCD), Feature Description.

Client Certificate Authentication

Using certificates for authentication can be considered more secure because a user cannot gain access to something simply by knowing the username and password. Using certificates prevents key loggers or other malware on a client machine from capturing keystrokes to identify user accounts and passwords.

The LoadMaster supports the use of certificates with KCD authentication. For more information refer to the Kerberos Constrained Delegation (KCD), Feature Description.

Dual-factor Authentication

Some authentication mechanisms assume a dual-factor approach where both the Active Directory and a secondary mechanism are used in sequence. For these, the form includes the username, password and also a passcode which is checked after the username and password.

OIDC OAUTH ESP Authentication

Open ID Connect (OIDC) is an identity layer added to the OAuth2.0 Protocol that enables authentication of users via tokens provided by an Identity Provider (IdP) (Referred to as the Authorisation Server role in Oauth). OIDC is commonly used to enabled Single Sign On of users across multiple applications via a single Identity Provider. OIDC uses the standardized message flows from OAuth2 to provide identity services.

When using OIDC, the loadmaster performs the Resource Server role, granting or denying access to an application via authorisation tokens. This requires an Identity Provider to be utilised for authenticating the users for example Microsoft Azure AD Identity Management.

For more information, refer to the OIDC OAUTH ESP Authentication, Feature Description.

8

Web Application Firewall Pack (WAF)

Web Application Firewall (WAF) services are natively integrated in the LoadMaster. This enables secure deployment of web applications, preventing Layer 7 attacks while maintaining core load balancing services which ensures superior application delivery performance, high availability and scalability. The WAF directly augments the LoadMaster's existing security features to create a layered defence for web apps, and therefore enables a safe, compliant and productive use of web application resources.

Some of the challenges faced when using a traditional firewall are listed in the section below. Some of the benefits of WAF are listed in the Challenges with Traditional Firewalls section.

Contents

- Challenges with Traditional Firewalls
- Benefits of WAF
- WAF-enabled LoadMaster Overview

Challenges with Traditional Firewalls

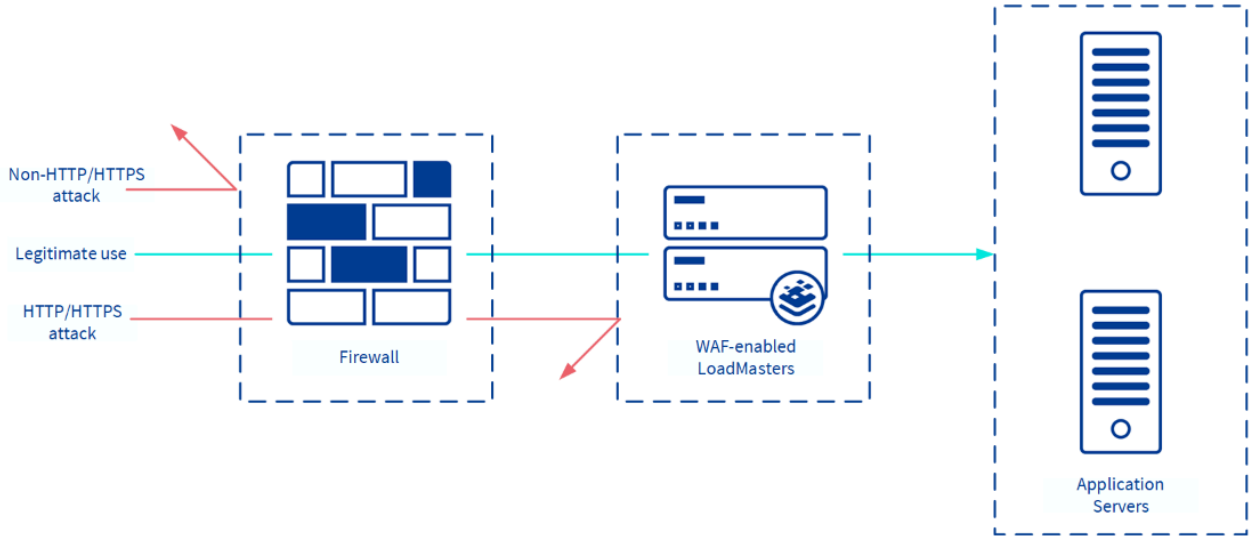
There are many limitations of traditional firewalls, such as the fact that they:

- Typically operate below Layer 4
- Have limited support for protocol inspection (that is, packet filtering)
- Are only aware of "ports and sockets"

Modern-day hacker attacks present many security challenges due to the fact that they are:

- Becoming increasingly complex
- Frequently changing their attack vectors
- Focusing on application vulnerabilities
- Harder to detect and block on a firewall

Benefits of WAF



As depicted in the diagram above, WAF offers many benefits, such as the fact that it:

- Processes web-based traffic, that is, HTTP/HTTPS
- Is able to detect modern application attack vectors
- Is deployed between perimeter defences and the application servers
- Works in conjunction with existing security technologies to provide complete application attack prevention

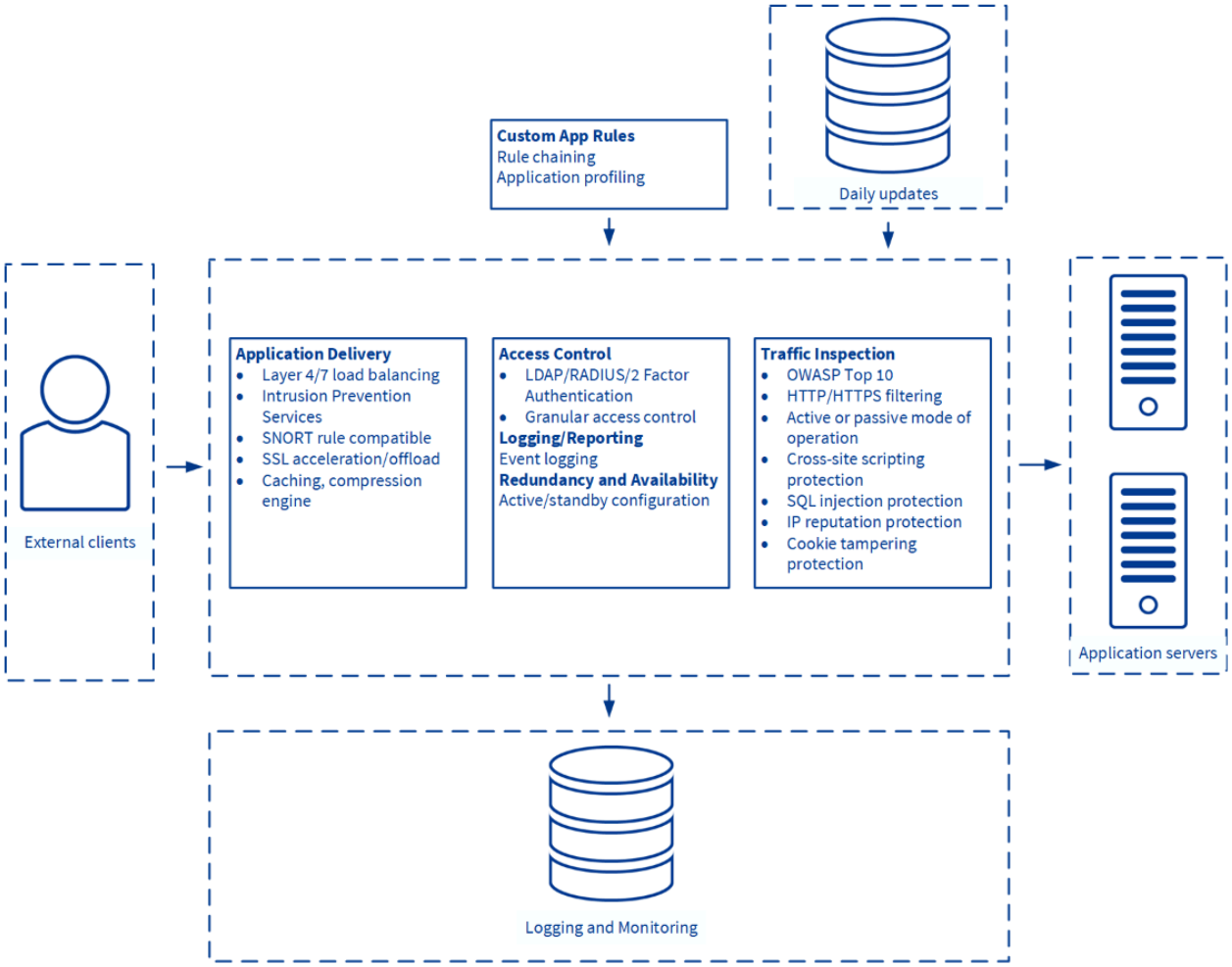
WAF-enabled LoadMaster Overview

Progress Kemp's integrated L7 WAF platform is based on an industry-leading rules engine that provides real-time coverage for all published application threats, including the Open Web Application Security Project (OWASP) Top Ten, as well as critical baseline protections. It also allows you to use your existing rules in a centralized Progress Kemp Application Delivery Controller (ADC) appliance.

With the WAF-ADC integration, Progress Kemp enhances its existing security features available on LoadMaster platforms, which currently feature edge security gateway functionality to protect published workloads including reverse proxy, Single Sign On (SSO), pre-authentication and SMTP domain filtering, dual-factor authentication, Intrusion Prevention System (IPS) and SSL bridging for secured traffic flows. Progress Kemp's commercial rules also includes attack categories such as IP reputation, botnet attack detection, web-based malware detection, webshell/backdoor detection, and HTTP Denial of Service (DoS) attack detection.

The LoadMaster includes WAF features and security services. This gives network administrators the ability to:

- Secure applications with minimum network impact and maximum protection
- Simplify infrastructure with this integrated solution
- Centralize required Application Delivery Controller (ADC) and WAF SSL overlay services into one product



As depicted in the diagram above, WAF provides a number of useful features such as:

- Fully-featured load balancing and content switching
- Intrusion Prevention System (IPS) and content filtering
- Protection against the Open Web Application Security Project's (OWASP) top ten vulnerabilities
- Support for standard and custom applications
- Active (block and log) mode operation support
- Passive (log only) mode operation support
- SQL injection protection
- Cross-site scripting mitigation
- Cross-Site Request Forgery (CSRF) prevention

- Cookie or form tampering prevention
- Distributed Denial of Service (DDOS) mitigation
- Trojan protection
- IP reputation checking
- Data leakage protection
- Built in reporting
- Built in logging including log field masking (that is, credit card numbers)

For further information, including steps on how to configure WAF, refer to the Web Application Firewall (WAF), Feature Description.

9

GEO

GEO assures seamless failover and failback to the best performing and geographically closest datacenter for optimal use of web-based applications, including Microsoft Exchange. In the event of a service disruption, traffic is automatically controlled based on a set policy in order to minimize impact and the need for manual intervention.

The GEO product is available in two forms:

- A standalone GEO product
- A Global Sever Load Balancing (GSLB) Feature Pack that is part of the LoadMaster product

GEO offers many load balancing algorithms including **round robin**, **weighted round robin**, **fixed weighting**, **real server load**, **location based** and **proximity**. “Round Robin” load balancing can be used for all active data centers, which includes support for weights and a chained failover option for disaster recovery. **Location Based** load balancing allows GEO to direct a client to a data center based on the client's country or continent, as defined by the created policies. **Proximity** takes **Location Based** one step further and allows for longitude and latitude granularity for definition of proximity. GEO securely and seamlessly integrates with LoadMaster to offer “Real Server Load” load balancing, in which GEO uses local data center metrics provided by LoadMaster, allowing clients to connect to the least busy data center.

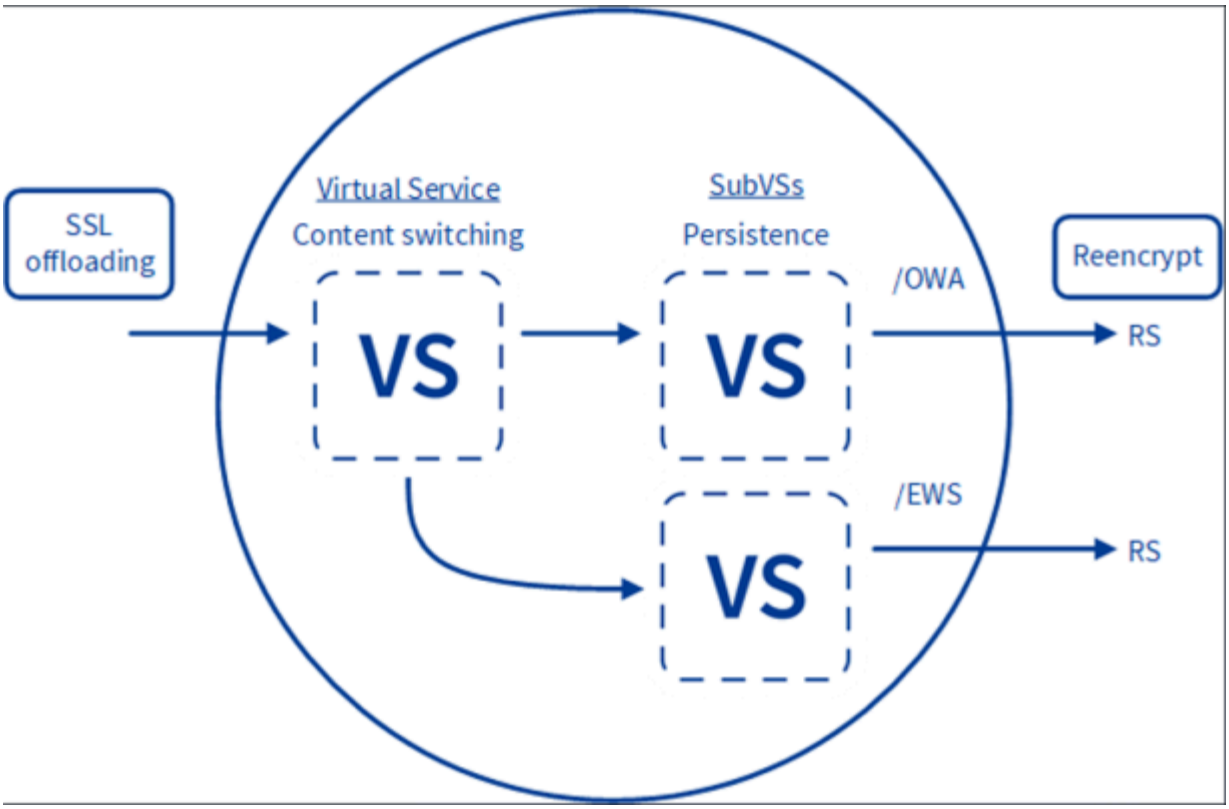
GEO can be deployed in a distributed (Active/Active) high availability configuration, with multiple appliances securely synchronizing information. Introducing GEO in your existing Authoritative Domain Name Services (DNS) requires minimal integration work and risk, allowing you to fully leverage your existing DNS investment.

For further information on GEO, please refer to the GEO Feature Description.

10

Sub-Virtual Services (SubVSs)

From within a Virtual Service you can create one or more ‘Sub-Virtual Services’ (SubVSs). A SubVS is linked to, and uses the IP address of, the ‘parent’ Virtual Service. The SubVSs may have different settings (such as health check methods, content rules etc.) to the parent Virtual Service and to each other. This allows the grouping of related Virtual Services, all using the same IP address. This could be useful when there are complex configurations such as Exchange which typically are comprised of a number of Virtual Services.



There are many advantages to using SubVSs, such as:

- SubVSs are linked to, and use the IP address of, the ‘parent’ Virtual Service
- Using SubVSs reduces the number of IP addresses required by applications such as Lync or Exchange
- Virtual Service to SubVS traffic is always transparent (no changes are required). SubVS to Real Server traffic can optionally be transparent (if it is transparent, the rules for transparency must be followed). Restrictions for using transparency still apply, for example, you cannot use transparency with re-encryption.
- SubVSs may have different settings (such as content rules) to the parent Virtual Service and to each other
- Using a SubVS provides the ability to have content switching and persistency on the same Virtual Service
- Using a SubVS gives the ability to perform multiple health checks on the same Virtual Service
- SubVSs work well with ESP, but ESP is not required

11

Certificates

Refer to the following sections for details on certificates.

Contents

- Self-Signed Versus CA-Signed Certificates
- Certificate Basics
- Operational Differences
- ACME Certificates

Self-Signed Versus CA-Signed Certificates

An SSL certificate is required for all SSL transactions, and as such is required for all SSL-enabled Virtual Services. With the LoadMaster, there are two types of SSL certificates:

- Self-signed certificates generated by the LoadMaster itself
- Certificates that are signed by a CA (Certificate Authority) such as Verisign or Thawte

When an SSL-enabled Virtual Service is configured on the LoadMaster, a self-signed certificate is installed automatically.

Generally, self-signed certificates should not be used for public-facing production websites.

They may be acceptable for use in some other scenarios, such as:

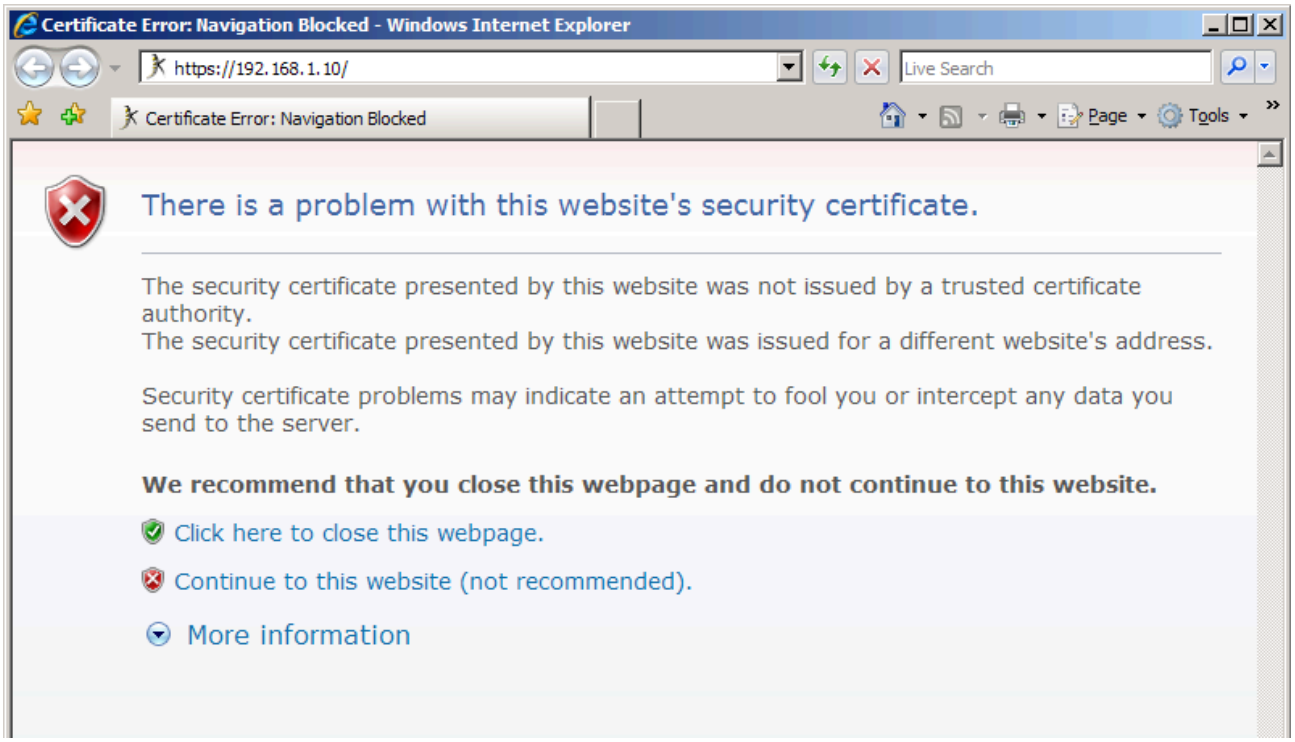
- Intranet sites
- QA sites, where web sites are tested but not presented to the general public

Certificate Basics

Both self-signed and CA signed certificates provide encryption for data in motion. A CA-signed certificate also provides authentication - a level of assurance that the site is what it reports to be, and not an impostor website.

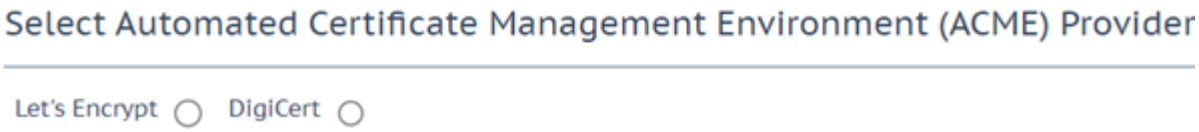
Operational Differences

The primary operational difference between a self-signed certificate and a CA certificate is that with self-signed, a browser will generally give some type of error, warning that the certificate is not issued by a CA. An example of the self-signed certificate error is shown in Figure 1.



This is the same warning message you receive when connecting to the LoadMaster WUI, as the WUI uses a self-signed certificate. Generally, this warning should occur only once per browsing session.

ACME Certificates



The LoadMaster provides an option of two Automated Certificate Management Environment (ACME) providers:

- Let's Encrypt
- DigiCert

Refer to the relevant section below for further details.

For detailed information related to DigiCert, refer to the DigiCert Feature Description.

Contents

- Let’s Encrypt Certificates
- DigiCert Certificates

Let’s Encrypt Certificates

Let's Encrypt is a free, automated, and open Certificate Authority (CA). It is a service provided by the Internet Security Research Group (ISRG).

Digital certificates are issued to enable HTTPS (SSL/TLS) for websites for free in a user-friendly way. The key principles for Let's Encrypt are:

- **Free:** Anyone who owns a domain name can use Let's Encrypt to obtain a trusted certificate at zero cost
- **Automatic:** Software running on a web server can interact with Let's Encrypt to painlessly obtain a certificate, securely configure it for use, and automatically take care of renewal
- **Secure:** Let's Encrypt serves as a platform for advancing TLS security best practices, both on the CA side and by helping site operators to properly secure their servers
- **Transparent:** All certificates issues or revoked are publicly recorded and available for anyone to inspect
- **Open:** The automatic issuance and renewal protocol is published as an open standard that others can adopt
- **Cooperative:** Much like the underlying internet protocols themselves, Let's Encrypt is a joint effort to benefit the community beyond the control of any one organization

Progress Kemp enables you to leverage the value of Let’s Encrypt certificates by automating the renewal and updating of certificates across your applications.

This includes:

- Support for HTTP-01 domain validation method
- Key generation
- Certificate issuance (create Certificate Signing Request (CSR) and request certificates)
- Automatic/manual certificate renewal and automatic updating of renewed certificates on the LoadMaster

DigiCert Certificates

DigiCert is the global leader in providing high-assurance TLS/SSL certificates and automated solutions to manage and deliver trust across all Public Key Infrastructure (PKI), Internet of Things (IoT), and signing workflows.

As of LoadMaster firmware version 7.2.58, Progress Kemp enables you to leverage the value of DigiCert certificates by automating the renewal and updating of certificates across your applications.

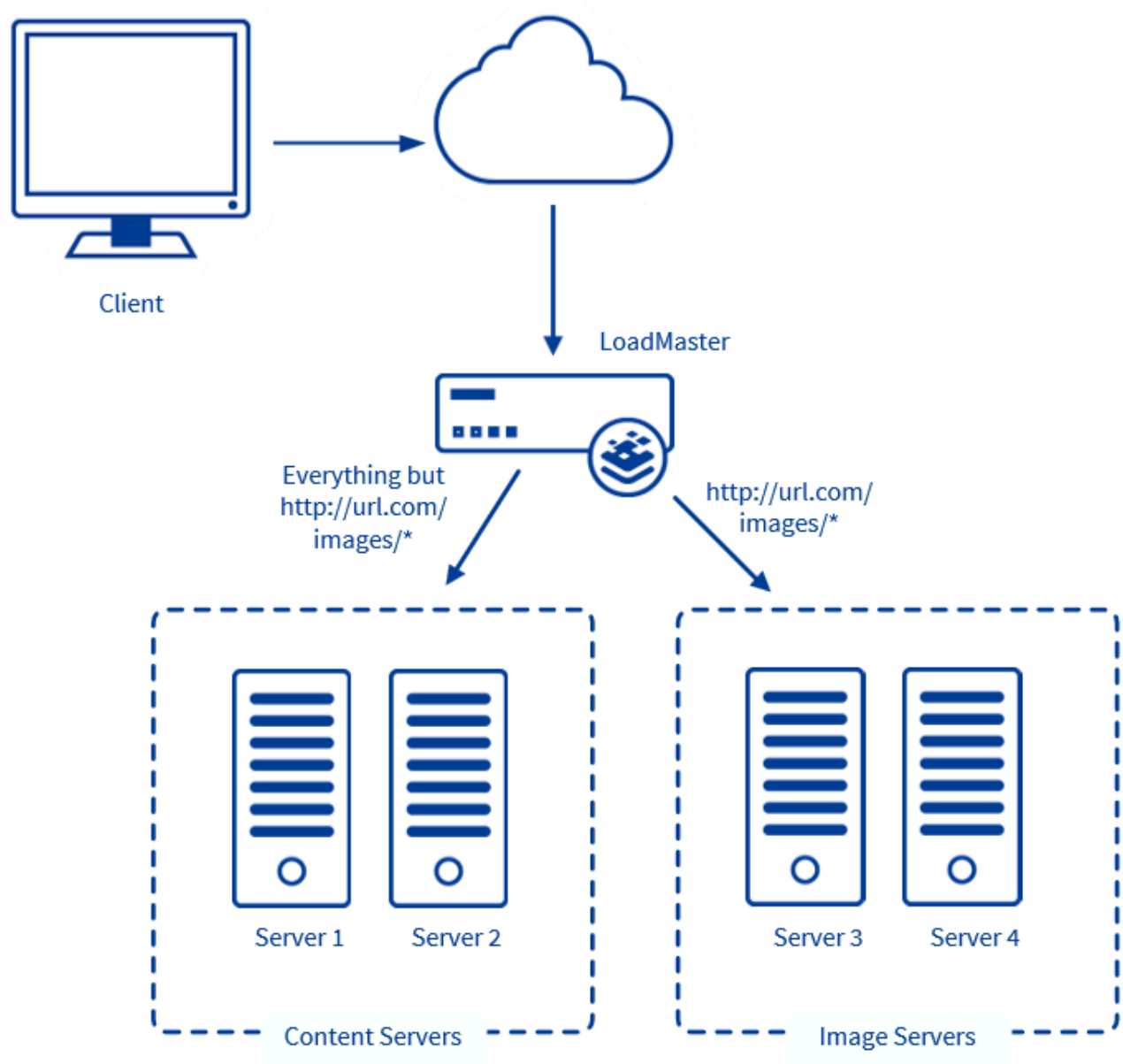
This includes:

- Support for the HTTP-01 domain validation method
- Key generation
- Certificate issuance (creation of Certificate Signing Request (CSR) and request certificates)
- Automatic/manual certificate renewal and automatic updating of renewed certificates on the LoadMaster

12

Rule-Based Content Switching

The LoadMaster series of load balancers support content switching, which is sometimes referred to a URL switching. This allows the LoadMaster to direct specific requests to specific Real Servers based on the contents of the requested URL.



For example, if you have two groups of servers; one group to serve images and the other to serve all other content - you can create content rules to separate these two classes of request.

Any URL that includes /images in it, such as “http://url.com/images/party.jpg” or “http://url.com/images/dogs.jpg” would be directed to server 3 and 4, while anything else would be directed to server 1 and 2.

This can be very useful if you have servers that perform different functions (application servers, static content servers, mapping servers, specialized content generation servers, etc.) that must all be served from the same general hostname (for example, www.websitename.com).

It is also possible to add header modification rules. For further details, refer to the Content Rules Feature Description.

Terminology

The term content switching does not refer to the process involved with Layer 2 switching. Instead, content switching refers to switching traffic between different servers, depending upon the content requested.

Using Content Switching

There are two parts to configuring content switching: the content rules, and the Virtual Service configuration. The content rules are configured globally on the LoadMaster and various rules are applied to specific Real Servers or SubVSs operating under a Virtual Service.

13

Health Checking

Refer to the sections below for details on LoadMaster health checks.

Contents

- Overview
- Service and Non-Service Based Health Checking

Overview

The LoadMaster utilizes health checks to monitor the availability of the Real Servers and the Virtual Services. If one of the servers does not respond to a health check within a defined time interval for a defined number of times, the weighting of this server will be reduced to zero. This zero weighting has the effect of removing the Real Server from the Virtual Service configuration until it can be determined that this Real Server is back online. The LoadMaster uses health checks that can be specified in the WUI. By default, the highest possible health check is associated with a Virtual Service. The LoadMaster performs Layer 7 health checks for the following ports:

Service	Port	Protocol
FTP	21	TCP

Service	Port	Protocol
TELNET	23	TCP
SMTP	25	TCP
HTTP	80	TCP
HTTPS	443	TCP
POP3	110	TCP
NNTP	119	TCP
IMAP	143	TCP
DNS	53	UDP
LDAP	389/636	UDP/TCP

Note: When creating a Virtual Service and using a service type other than Generic, additional health checking protocols are available. For example, the service type Remote Terminal will permit checking with Remote Terminal Protocol.

Note: The Remote Terminal Protocol supports Network Level Authentication.

For other ports the LoadMaster uses Layer 4 health checks for TCP services and Layer 3 health checks for UDP services. The settings for the health checks can be changed from the default settings using the Virtual Service wizard to accommodate non-standard settings. For example, one could run an http service on port 8080 instead of 80, and change the health check to HTTP instead of the default Layer 4 check.

Note: These global settings hold for all servers in the farm, that is, you cannot assign different timeouts for different servers.

It is mandatory that one of the service checking options be used when defining a Virtual Service on the LoadMaster.

Service and Non-Service Based Health Checking

Layer 3 health checks utilize ICMP based echo requests (pings) to test whether a Real Server can be reached over the network. A Layer 3 check is not Virtual Service specific, for example, when it fails, the corresponding Real Server will be removed from all Virtual Services that use it.

In contrast to the Layer 3 health checks, service based health checking for both the Layer 4 and Layer 7 health checks are Virtual Service based. When a Real Server fails such a check, it will be removed only from the corresponding Virtual Service – all other Virtual Services that use this Real Server are unaffected.

Type	Description
ICMP	The LoadMaster sends ICMP echo requests (pings) to the Real Servers. A Real Server fails this check when it does not respond with an ICMP echo response in the configured response time for the configured number of retries.
TCP	The LoadMaster attempts to open TCP-connection to the Real Server on the configured service port: it sends a TCP SYN packet to the server on the service port. The server passes the check if it responds with a TCP SYN ACK in the response time interval. In this case the LoadMaster closes the connection by sending a TCP RESET. If the server fails to respond within the configured response time for the configured number of times, it is assumed dead.
FTP	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 21). If the server responds with a greeting message with status code 220, the LoadMaster sends a QUIT command to the server, closes the connection and marks it as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different status code, it is assumed dead.
TELNET	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 23). If the server responds with a command string beginning with the char ‘0xff’, the LoadMaster closes the connection and marks the server as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different command string,

Type	Description
	it is assumed dead.
SMTP	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 25). If the server responds with a greeting message with status code 220, the LoadMaster sends a QUIT command to the server, closes the connection and marks it as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different status code, it is assumed dead.
HTTP	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 80). The LoadMaster sends a HTTP/1.0 HEAD request the server, requesting the page "/". If the server sends a HTTP response with a status code of 2 (200-299, 301, 302, 401) the LoadMaster closes the connection and marks the server as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different status code, it is assumed dead. HTTP 1.0 and 1.1 support available. HTTP 1.1 allows you to check host header enabled web servers.
HTTPS	The LoadMaster opens an SSL connection to the Real Server on the Service port (port 443). The LoadMaster sends a HTTP/1.0 HEAD request the server, requesting the page "/". If the server sends a HTTP response with a status code of 2 (200-299, 301, 302, 401) the LoadMaster closes the connection and marks the server as active. If the server fails to respond within the configured response time for the configured number of times or if it responds with a different status code, it is assumed dead. HTTP 1.0 and 1.1 support available. HTTP 1.1 allows you to check host header enabled web servers.
POP3	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 110). If the server responds with a greeting message that starts with +OK, the LoadMaster sends a QUIT command to the server, closes the connection and marks it as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a

Type	Description
	different status code, it is assumed dead.
NNTP	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 119). If the server responds with a greeting message with status code 200 or 201, the LoadMaster sends a QUIT command to the server, closes the connection and marks it as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different status code, it is assumed dead.
IMAP	The LoadMaster opens a TCP connection to the Real Server on the Service port (port 143). If the server responds with a greeting message that starts with "+ OK" or "* OK", the LoadMaster sends a LOGOUT command to the server, closes the connection and marks it as active. If the server fails to respond within the configured response time for the configured number of times, or if it responds with a different status code, it is assumed dead.
DNS	The **Name Server (DNS) Protocol** value is only available in the **Real Server Check Method** drop-down list when the Virtual Service **Protocol** is set to **udp**. The LoadMaster performs nslookups against an A record on the server over UDP port 53. If the server successfully responds to the DNS query, the LoadMaster marks it as active. If the server fails to respond within the configured response time for the configured number of times or if it responds unsuccessfully to the A record, it is assumed down.
RDP	The LoadMaster sends an RDP Routing Token to the Real Server. The RDP health check supports Network Level Authentication.
Binary	Specify a hexadecimal string to send to the Real Server. Specify a hexadecimal string which will be searched for in the response sent back from the Real Server. If the

Type	Description
	LoadMaster finds the pattern in the response, the Real Server is considered up. Specify the number of bytes to search for the reply pattern within.
LDAP	Select an LDAP endpoint to use for the health check. The LDAP health check uses the LDAP credentials and protocol specified in the LDAP endpoint. The health check is run against the Real Server IP address and port. The LDAP health check comprises of a LoadMaster connecting to a Real Server and validating the specified user credentials. The health check is performed in two steps: Step 1: Check if the Real Server specified port is up and available. Step 2: Attempt to log in to the Real Server using the LDAP specified credentials. If step 1 and step 2 are true, the health check passes. If step 1 or step 2 fails, the health check fails.
None	No health checking is performed.

14

SNMP Support

Simple Network Management Protocol (SNMP) is a protocol that enables the management of many network devices over the network from a remote management station (SNMP manager).

The manager station can request data from the managed stations (SNMP agents) or it can change the value of data on the agents.

The managed stations (SNMP agents) can also be set up to alert the manager when some predefined events occur, such as a unit failover. The alerting mechanism uses event traps.

The current version is SNMPv3. The two previous revisions in use are SNMPv1 and SNMPv2c (community-based SNMPv2).

The SNMP support of the LoadMaster is based on SNMPv3, and is backwards-compatible such that all three of the above versions can be used. However, since SNMPv1 does not support 64bit-values (as used in the LoadMaster Management Information Base (MIB)), it is recommended to use SNMPv2c or SNMPv3. MsgSecurity is supported only with SNMP v1 and v2c.

Note: When monitoring the LoadMaster in HA mode please monitor individual appliances by the appropriate Ethernet address.

The information regarding all LoadMaster-specific data objects is stored in the enterprise-specific MIBs listed below.

ONE4NET-MIB.txt	enterprise id
IPVS-MIB.txt	Virtual Server stats
B-100-MIB.txt	LoadMaster configuration data
CERTS-MIB.txt	SSL certificate information

These MIBs need to be installed on the SNMP manager machine in order to be able to request the performance/config-data of the LoadMaster using SNMP.

SNMP is supported on both IPv4 and IPv6 Virtual Services at both Layer 4 and Layer 7.

The SNMP support is disabled by default.

You can retrieve SSL certificate information including the file name, certificate Subject Name, certificate serial number, certificate start date, certificate end date, and certificate issuer information using SNMP. SNMP can display this information for up to 256 SSL certificates.

You can also retrieve disk space usage details using SNMP. The /var/log and /var/log/userlog partition information is available.

15

LoadMaster Software Upgrades

Refer to the following section for details on upgrading the LoadMaster software.

Online Upgrades

The LoadMaster provides the ability to perform online software updates and upgrades. Patches will be made available by Progress Kemp. These patches should be installed on a machine which supports an FTP, HTTP or SSH daemon.

Patches are checksummed (with MD5) and encrypted to protect against data corruption or tampering.

Patches can be installed using one of two methods:

- Using the Console Line Interface (CLI)
- Using the WUI

Using the CLI

Using the Configuration menu, select the **Utilities > Software Upgrade** menu option.

When the patch has been downloaded, the patch will be unpacked and checked.

If the patch is valid, the patch version will be displayed and the user will be asked if the patch should be installed.

Upon the successful installation of the patch, the LoadMaster should be rebooted to activate the new version.

Using the WUI

To upgrade the software using the WUI, follow the steps below:

1. In the main menu of the LoadMaster WUI, select System Configuration > System Administration > Update Software.
2. Click the **Browse…** button and browse to and select the patch.
3. Click the **Update Machine** button.

Upon successful installation of the patch, the LoadMaster should be rebooted to activate the new version.

If the patch does not install correctly, the previous version of the software may be reactivated using the configuration menu or WUI.

License information can be entered in the WUI as described in the Licensing, Feature Description document.

After updating a license key, a reboot should be performed to enable the new functionality.

Patch support can expire; if this happens you will be notified during the upgrade procedure.

Note: If you receive the message 'Update not permitted' please contact Progress Kemp for re-licensing.

16

User Management

The LoadMaster supports multiple user logins with varying levels of access that can be managed by navigating to **System Configuration > System Administration > User Management**. Each username must be a minimum of three characters and a maximum of fourteen. Passwords must be a minimum of six characters long. Users created here can only access the WUI, remote access using SSH is not supported.

The LoadMaster can be configured to utilize RADIUS server or client certificates in the authentication process.

For further information on User Management in general, and client certificate WUI authentication, refer to the User Management, Feature Description.

For further information on RADIUS WUI authentication, refer to the RADIUS Authentication and Authorization, Technical Note.

Roles/Permission

The factory default username is **bal** and the default password is **1fourall**. The factory default user retains the highest level of access. All users created have a subset of the access permitted by the default account. Changing roles for users takes effect in real-time. Roles can be combined and are mutually exclusive.

The default access for users is read only access to the LoadMaster WUI, generating Certificate Signing Requests (CSRs), read access to log files and the ability to perform basic debugging.

Real Servers

This role permits enabling and disabling Real Servers.

Note: Users with the Real Servers permission cannot add SubVSs.

Virtual Services

This role permits managing Virtual Services. This includes SubVSs. Virtual Service modifications permitted include add, delete and modify for any subnet.

Rules

This role permits managing Rules. Rule modifications permitted include add, delete and modify.

System Backup

This role permits performing system backups.

Certificate Creation

This role permits managing SSL Certificates. Certificate management includes adding, deleting and modifying SSL Certificates.

Intermediate Certificates

This role permits managing intermediate Certificates. Certificate management includes the ability to add and delete intermediate certificates.

Certificate Backup

This role permits the ability to export and import certificates.

User Administration

This role is allowed access to all functionality within the **System Configuration > System Administration > User Management** screen.

All Permissions

This role gives users all permissions except the permission to change the bal password and the permission to create or delete other users.

GEO Control

This role is used only with the LoadMaster GEO product. For more information on GEO and the Global Server Load Balancing (GSLB) Feature Pack, refer to the GEO Feature Description.

17

WUI Authentication and Authorization

The **WUI Authentication and Authorization** screen enables the administration of the available authentication (login) and authorization (allowed permissions) options.

Authentication

Users must be authenticated before logging on to the LoadMaster. The LoadMaster allows authentication of users to be performed using the RADIUS and LDAP authentication methods as well as Local User authentication.

When all authentication methods are selected, the LoadMaster attempts to authenticate users using the authentication methods in the following order:

1. RADIUS
2. LDAP
3. Local Users

For example, if the RADIUS server is not available then the LDAP server is used. If the LDAP server is also not available then Local User authentication methods are used.

If neither RADIUS nor LDAP authentication methods are selected, then the Local User authentication method is selected by default.

The consequence of this ordering is that when a local user (that is using a local password) logs in to the LoadMaster and they do not exist in the RADIUS or LDAP configuration, error messages appear in the log for the RADIUS and LDAP authentication checks, even though local password authentication succeeds.

Users must be authenticated before logging on to the LoadMaster. The LoadMaster allows authentication of users to be performed using the LDAP authentication method as well as Local User authentication.

Authorization

LoadMaster allows the users to be authorized by either RADIUS or using Local User authorization. The user's authorization decides what level of permissions the user has and what functions on the LoadMaster they are allowed to perform.

Note: You can only use the RADIUS authorization method if you are using the RADIUS authentication method.

When both authorization methods are selected, the LoadMaster initially attempts to authorize the user using RADIUS. If this authorization method is not available, the LoadMaster attempts to authorize the user using the Local User authorization. Authorization using LDAP is not supported.

If the RADIUS authorization method is not selected, then the Local User authorization method is selected by default.

Below is an example of the configuration that needs to be on the radius server for authorization to work.

Note: The below example is for Linux only.

The Reply-Message should be self-explanatory on what permission it's allowing. They should match up to the WUI's user permissions page, with the exception of "All Permissions":

LMUSER Cleartext-Password := "1fourall"

Reply-Message = "real,vs,rules,backup,certs,cert3,certbackup,users"

Note: The **bal** user is always authenticated and authorized using the Local User authentication and authorization methods.

18

Bonding and VLAN

Refer to the following sections for details about the LoadMaster bonding/VLAN functionality.

Overview

LoadMaster bonding/VLAN tagging can be easily set up and configured using the WUI. Successful deployment requires that the pre-requisites have been satisfied. This guide is designed to introduce interface bonding and VLAN configuration on the LoadMaster. Bonding support is available with all network modules.

Contents

- Prerequisite (Switch Compatibility)
- Bonding/Teaming (802.3ad/Active-Backup)
- VLAN Tagging

Prerequisite (Switch Compatibility)

A list of prerequisites are below:

- VLAN Tagging
- IEEE 802.1Q
- Bonding/Teaming (802.3ad/Active-Backup)
- IEEE 802.1AX/IEEE 802.3ad/LACP

Switch Configuration

Enabling the Active-Backup mode generally does not require switch intervention and can be configured directly on the LoadMaster. Using the 802.3ad bonding mode will require configuring a link aggregation group on the switch in conjunction with the LoadMaster. Please read the switch documentation to establish the corresponding team/bond, common terms for link aggregation, which include; "Ethernet trunk", "NIC teaming", "port channel", "port teaming", "port trunking", "link bundling", "EtherChannel", "Multi-Link Trunking (MLT)", "NIC bonding", "Network Fault Tolerance (NFT)" and “LAG”.

When enabling VLAN trunking on the switch port make sure to configure the port to support the appropriate mode; General, Access, or Trunking. General descriptions are as follows (check the switch documentation for specifics):

- General: the port belongs to VLANs, and each VLAN is user-defined as tagged or untagged (full 802.1Q mode)
- Access: the port belongs to a single untagged VLAN
- Trunk: the port belongs to VLANs in which all ports are tagged

Bonding/Teaming (802.3ad/Active-Backup)

There are a few key things to keep in mind when creating bonds/teams:

- You can only bond interfaces higher than the parent, so if you choose to start with port 10 then you can only add ports 11 and greater
- Bond links first if you need VLAN tagging, then add VLANs after the bond has been configured
- In order to add a link to a bonded interface, any IP addressing must first be removed from the link to be added
- Enabling the Active-Backup mode generally does not require switch intervention
- Bonding eth0 with eth1 can lead to serious issues and is not allowed to occur

Note: Ensure that all bonded interfaces are configured for the same link speed, both on the switch and LoadMaster.

Note: If bonding port 0, we recommend you move the web administrative interface and/or the remote SSH access to a different port temporarily until the bonding has been completely configured and is working.

VLAN Tagging

Things to keep in mind:

- Configure VLAN tagging on the switch first, if required
- Start by deciding if bonding is required. If it is, first establish your bonded configuration and then proceed by adding the VLAN tagging information
- VLANs can be added to physical interfaces or bonded interfaces

19

IPsec Tunneling

Internet Protocol Security (IPsec) is designed and used to provide secure connections between nodes and networks throughout the internet. IPsec has become the standard for most of the IP Virtual Private Network (VPN) technology.

IPsec can operate in a point-to-point (aka host-to-host) configuration or in a site-to-site (aka network-to-network) configuration. An IPsec implementation operates in a host, as a Security Gateway (SG), or as an independent device, affording protection to IP traffic for both IPv4 and IPv6. (A security gateway is an intermediate system implementing IPsec, for example a firewall, router or gateway which has been IPsec-enabled.)

There are many benefits of using IPsec. These include, but are not limited to:

- Secure connectivity provided across distributed enterprises
- Bandwidth benefits over traditionally expensive Wide Area Network (WAN) infrastructure
- Cost benefits over traditionally expensive WAN infrastructure
- Security - IPsec VPNs inherently provide a high degree of data security
- Flexibility - IPsec VPNs can be established and be available using the internet
- Resilience and High Availability (HA) for critical and sensitive applications available over the internet

For further information, including step-by-step instructions on how to configure the LoadMaster to work with IPsec tunneling, refer to the IPsec Tunneling, Feature Description.

20

Miscellaneous

Refer tot he following sections for details on some miscellaneous LoadMaster features.

Contents

- IPv6 Support
- Remote Syslog Support
- How to Get a License
- Backup and Restore
- Disable/Enable Access to the WUI
- Interoperability Between L4 and L7 Virtual Services
- Log Information
- Debugging Utilities
- RESTful API Interface

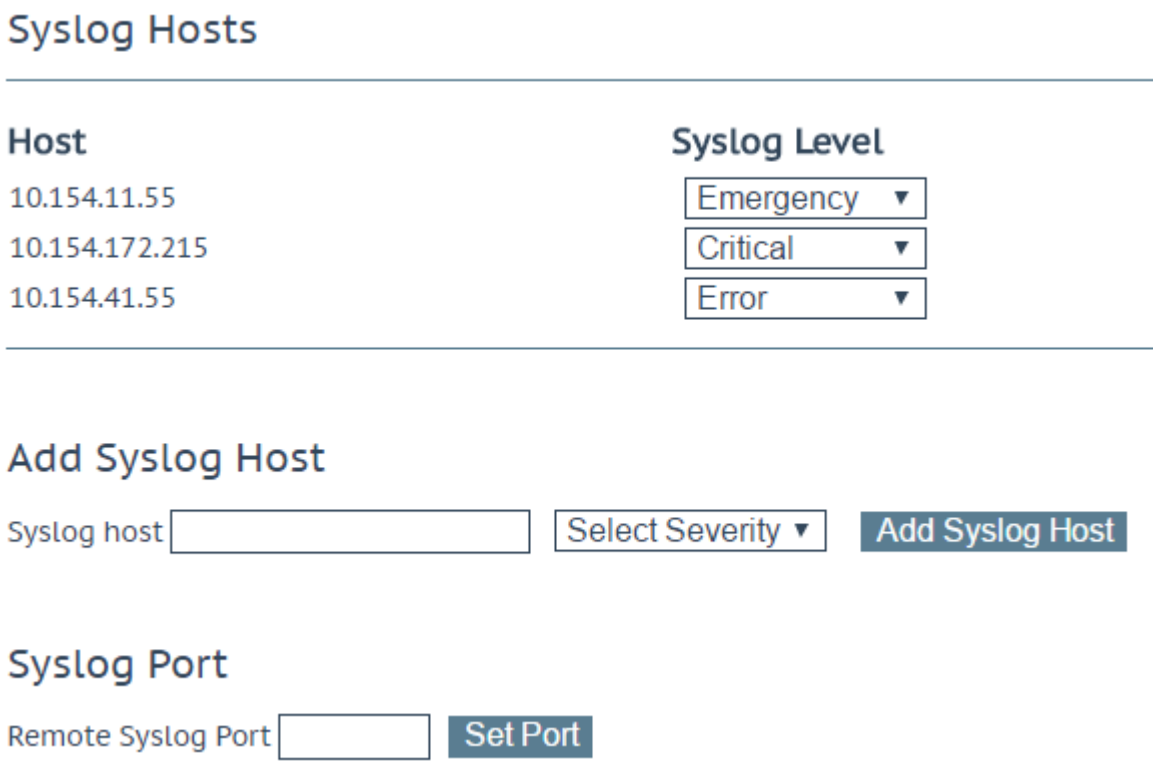
IPv6 Support

This version of LoadMaster software includes IPv6 support at both Layer 4 and Layer 7. Before you lay out your network addresses, consider which will remain as IPv4 and which will convert to IPv6. The LoadMaster is capable of supporting, and is able to translate between IPv4 and IPv6, unlike networks. Therefore, you may have an internal network that is IPv6, and interconnect to an external IPv4 network.

Note: FTP for IPv6 at Layer 4 is not supported.

Remote Syslog Support

The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally.



It is also possible to configure the LoadMaster to transmit these error messages to a remote syslog server by entering the relevant IP address in the **Syslog host** text box, selecting the severity and clicking **Add Syslog Host**. Six different error message levels are defined. Each level of message may be sent to a different host server.

Notice messages are sent for information only. Emergency messages normally require immediate user action.

Note: To enable a syslog process on a remote Linux server to receive syslog messages from the LoadMaster, the syslog must be started with the "-r" flag.

How to Get a License

To unlock the LoadMaster software, you need a license. The license will be generated individually for each LoadMaster instance in conjunction with an Access Code.

There are three different licenses that you can get for a LoadMaster:

- A trial license - this is a fully functional license which is valid for up to 30 days
- A full, non-time-limited LoadMaster license
- A full, non-time-limited license for a LoadMaster High Availability (HA) cluster consisting of two machines

A trial license can be upgraded to either a full single or a full HA license.

License information can be updated in the Web User Interface at **System Configuration** > **System Administration** > **License Management**. Repeat the process for a second LoadMaster if using HA systems. We recommend rebooting after updating the license.

You need a Progress ID to license a Virtual LoadMaster for the first time. If you do not have a Progress ID, refer to the Licensing, Feature Description, for instructions on how to set one up. When you have a Progress ID, you can license the Virtual LoadMaster using the WUI options provided.

If you need to upgrade a license, contact Progress Kemp.

Backup and Restore

You can backup and restore the LoadMaster configuration settings as needed. You can take manual backups, but you can also save backups to a remote server. The complete configuration (the Virtual Service, GEO, ESP and base configuration) of the LoadMaster is saved to a single file on the server along with statistical data.

Note: No SSL certificate information is contained within a backup.

The server must be running an FTP daemon or an SSH daemon. By default the remote protocol is FTP but that can be changed to SCP.

When restoring a configuration, you specify what parts of the configuration should be restored:

- The LoadMaster base configuration only
- The Virtual Service configuration only
- The GEO configuration only
- The ESP SSO configuration only
- A combination of the Virtual Service, GEO, ESP, and/or LoadMaster base configuration

The base configuration contains the information about the basic configuration of the LoadMaster, that is, the IP addresses of the various interfaces and the keyboard and time zone settings.

The Virtual Service configuration contains only the settings relating to the Virtual Services and the Real Servers.

The GEO configuration contains only the settings relating to the GEO configuration.

The ESP SSO configuration stores the SSO domains, LDAP endpoints and SSO custom image sets. This

does not restore the Virtual Service settings - use the **VS Configuration** option to restore those.

Note: Prior to restoring your configuration, please ensure that the LoadMaster receiving the configuration is in the same High Availability (HA) mode as when the backup was taken. The LoadMaster Base Configuration currently only allows migration from an individual unit to an individual unit or from HA to HA. This cannot currently cross HA configuration types. For more information on how to set up HA, refer to the Set Up HA section of the High Availability (HA) Feature Description.

You can configure automated backups on a daily or weekly basis.

Disable/Enable Access to the WUI

In certain circumstances users may wish to disable access to the LoadMaster WUI. This can be done by going to **Certificates & Security > Remote Access** in the LoadMaster WUI and deselecting the **Allow Web Administrative Access** check box.

Note: It is critical to the security of your appliance that you use a dedicated network interface for management traffic; refer to the LoadMaster Hardening Technical Note for further details.

If the access is disabled, no users will be able to use the LoadMaster WUI. This may be required in certain circumstances for security or other reasons.

Interoperability Between L4 and L7 Virtual Services

When a service is switched from one persistency method to another, the absolute values of all Virtual Service and Real Server counters will be reset to zero.

This may cause peaks in the service graphs when displaying relative values (bytes per second, etc.) when, for example, the bytes counter jumps from terabyte values to zero.

Log Information

Log files are viewable in the WUI at **System Configuration > Logging Options > System Log Files:**

- **Boot.msg File** contains Linux standard boot information
- **Warning Message File** contains a list of events generated by the core load-balancing engine
- **System Message File** contains a list of events generated by the core load balancing and the underling Linux operation system

---

Note: Log files are volatile. To ensure that critical log information is available in the event of a recycle on the LoadMaster, please use the syslog facility.

Debugging Utilities

Debugging utilities can be executed in the WUI at **System Configuration > Logging Options > System Log Files > Debug Options**. These are best utilized when working with the Progress Kemp Support Team.

Disable All Transparency

Disable transparency for all Virtual Services - this option should only be changed with approval from the Progress Kemp Support Team.

Enable L7 Debug Traces

Enable additional debugging information that is captured in the System Messages log.

Perform a PS

Reports the Process Status (PS).

Perform a l7adm

Display detailed information about the Layer 7 Virtual Services.

Ping Host

Issues an ICMP echo request to any IPv4 device. Please make sure the target IP supports ICMP.

RESTful API Interface

LoadMaster provides an interface designed to allow remote applications access in a simple and consistent manner. The interface is a REST-like interface. REST (REpresentational State Transfer) is a style of software architecture for distributed systems and is one of the predominant web service design models.

The LoadMaster RESTful API works by allowing a user or application to pass HTTP requests to the LoadMaster. The LoadMaster answers the request with an XML-formatted response.

The interface is disabled by default. For more information on enabling the interface, please refer to Web User Interface (WUI), Configuration Guide.

For further details on the RESTful API, please refer to the RESTful API documentation.

21

Network Telemetry

The LoadMaster can monitor the network traffic traversing its interfaces and generate rich network telemetry in IP Flow Information Export (IPFIX) format.

IPFIX is a flow export standard used to identify and collect application and transaction data in a network infrastructure. Flow data provides visibility into application traffic utilization and structure at any time, enabling you to report on key network performance metrics related to application workload. This is often leveraged as an alternative to full packet capture and analysis for ongoing monitoring of a network infrastructure. Various network devices including switches, firewalls, load balancers, and routers typically provide flow-based feeds to collectors which are then analyzed by a performance monitoring and analytics toolset.

Using Network telemetry functionality, the LoadMaster is now able to participate in providing flow data visibility in conjunction with a compatible IPFIX data analysis system. We recommend using the Progress Flowmon Collector for this data analysis. The Flowmon Collector is the ideal network monitoring appliance that captures, stores, and processes flow data, including normalization, visualization, and analysis.

For further details on Network Telemetry, refer to the Network Telemetry Feature Description.

22

References

Other useful documents are as follows:

Web User Interface (WUI) Configuration Guide

Configuring Real Servers for DSR Technical Note

Writing a Resource Based Adaptive Server Agent Technical Note

RESTful API Documentation

Oracle Virtual Box Installation Guide

Cisco UCS B-Series Blade Server Installation Guide

KVM (Para-Virtualized) Installation Guide

Hyper-V Installation Guide

VMware Installation Guide

LoadMaster Licensing Feature Description

SSL Accelerated Services Feature Description

Web Application Firewall (WAF) Feature Description

RADIUS Authentication and Authorization Technical Note

RSA Two Factor Authentication Feature Description

Kerberos Constrained Delegation (KCD) Feature Description

IPsec Tunneling Feature Description

SDN Adaptive Load Balancing Feature Description

User Management Feature Description

LoadMaster Clustering Feature Description