



LoadMaster 3620 SSL-Master 1020



Installation and Configuration Guide



Copyright © 2000 - 2005 KEMP Technologies, Inc. All rights reserved.
KEMP Technologies, Inc. reserves all ownership rights for the *LoadMaster* product line including software and documentation.

The use of the *LoadMaster Load Balancer* is subject to the license agreement.

Information in this guide may be modified at any time without prior notice.

Company names used in examples are fictitious unless otherwise noted.

Sun, Sun Microsystems, the Sun Logo, Solaris, SunOS and *Java* are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

UNIX is a registered trademark of X/Open Company Ltd.

IBM is a registered trademark of International Business Machines Corporation.

Microsoft, Windows and *Windows NT* are registered trademarks of Microsoft Corporation.

Linux is a registered trademark of Linus Torvalds.

Intel and *Pentium* are registered trademarks of Intel Corporation.

All other trademarks and registered trademarks are the property of their respective owners.

Document version: V1.4
Date of issue: 02/11/2005



Table of Contents

SECTION I. APPLICATION GUIDE	8
A. Preface	8
1. Foreword	8
2. LoadMaster vs. SSL-Master Feature/Function Comparison	8
3. Typographical Conventions	8
4. Glossary and Abbreviations	8
B. Overview of the LoadMaster	10
1. Load Balancing and its Benefits	10
2. Considerations in Getting Started	10
3. A Simple Balancer Configuration.....	12
4. LoadMaster Load Balancer Features	13
C. LoadMaster Network Topologies.....	14
1. One-Armed Balancer	14
2. Two-Armed and Multi-Armed Balancer	15
3. Direct Server Return – DSR example	16
D. Miscellaneous Networking Issues	17
1. S-NAT	17
2. Default Gateway and Routes.....	17
E. Single/Dual Unit Configurations	19
1. Single Unit Configuration.....	19
2. High Availability (HA) Configuration	19
F. Balancing Methods.....	20
1. Round Robin	20
2. Weighted Round Robin.....	21
3. Least Connection	21
4. Weighted Least Connection.....	21
5. Agent Based Adaptive Balancing.....	21
6. Fixed Weighting	22
G. Layer 4 Persistency	22
1. Source IP Address Based Persistency	22
H. Layer 7 Persistency	22
1. SSL session ID Based Persistency.....	22



2. URL Based Persistency	23
3. URL Host Based Persistency.....	23
4. Cookie Based Persistency	23
5. Cookie/Source (Cookie or IP source) Based Persistency	23
6. Active Cookie Based Persistency.....	23
7. Active Cookie/Source (Active Cookie or IP source) Based Persistency.....	23
8. Cookie hash Persistency.....	23
9. Port Following.....	23
I. SSL Acceleration	24
1. Reverse Acceleration	24
2. Certificate Files.....	24
3. 3 rd Party Certificates.....	25
J. Rule Based Content Switching.....	25
1. Rule Definition.....	25
1.1. Special Characters.....	25
1.2. Regular expressions.....	26
1.3. Host name matching.....	26
K. Health Checking	26
1. Service and non-service based Health Checking.....	27
L. SNMP Support	28
1. LoadMaster Performance Metrics via SNMP	29
2. LoadMaster Event Traps via SNMP	29
M. LoadMaster Software Upgrades	30
1. Online Upgrades	30
N. Miscellaneous.....	31
1. Remote Syslogd Support	31
2. How to get a license	31
2.1. Get a 30 day evaluation license	31
2.2. Get a full LoadMaster license.....	32
2.3. Get full High Availability LoadMaster cluster licenses.....	32
2.4. Upgrading the evaluation license to a full single or HA license	32
3. Backup and Restore	33
4. System recovery.....	33
5. Interoperability between L4 / L7 Virtual Services	33
O. Appendix I.....	33



1. API for Agent Based Adaptive Balancing	33
2. Http Server Configuration for Cookie Support	35
3. MIB-tree	36
II. INSTALLATION AND CONFIGURATION GUIDE	36
A. Before Getting Started.....	36
1. The LoadMaster Appliance	37
1.1. Delivery Content	37
1.2. LoadMaster 1500 Hardware	37
2. Connecting the Hardware	37
B. Initial Setup of your LoadMaster Single Unit (non-HA).....	37
1. Login and License Key	37
C. Initial Setup of a LoadMaster High Availability (HA) Cluster.....	38
1. Login and License Key	38
2. Configuring the second LoadMaster.....	38
D. Quick Setup.....	39
E. Main Menu	40
1. Configuration Menu basics.....	40
1.1. Quick Setup.....	41
2. Service Management (CLI)	41
3. Local Administration	41
3.1. Set Password.....	41
3.2. Set Date/Time	41
3.3. Set Keyboard Map	41
3.4. Backup/Restore	42
3.5. Remote Access Control.....	42
4. Basic Setup.....	42
4.1. Network configuration.....	42
4.2. Hostname Configuration	43
4.3. DNS configuration	43
5. Extended Configuration.....	43
5.1. Interface Control.....	43
5.2. Enable/Disable S-NAT	43
5.3. Syslogd Configuration	44
5.4. SNMP metrics	44
5.5. SNMP traps	44



5.6. Enable/Disable L7 persistency state failover	45
5.7. Enable/Disable L4 connection state failover	45
5.8. Multicast Configuration.....	45
5.9. HA timeout.....	45
6. Packet Filter & Access Control Lists.....	45
6.1. Access control Lists	45
7. Utilities.....	46
7.1. Software Upgrade	46
7.2. Transfer mode	46
7.3. Network Time Protocol Host.....	47
7.4. SSL certificate administration.....	47
7.5. Update License	47
7.6. Diagnostics	47
8. Reboot	48
9. Exit LoadMaster Config	48
F. The LoadMaster Questionnaire	48
1. Single LoadMaster Balancer Solution.....	48
2. Highly Available dual LoadMaster Balancer Solution.....	48
III. COMMAND LINE INTERFACE.....	49
2. Adaptive scheduling command level.....	50
3. Health check command level.....	51
5. Rule Edit command level.....	52
6. Virtual Service (VIP) command level.....	53
IV. WEB USER INTERFACE (WUI) CONFIGURATION GUIDE	58
A. Glossary and Abbreviations	58
B. Fast Track.....	59
1. How To Login	59
2. Create a Simple Virtual Service	59
3. Create a Virtual Service with Content Rules	62
4. Create an SSL accelerated Virtual Service.....	65
C. Full Menu Tree	66
1. Home.....	67
2. Virtual Services	67
2.1. Add Virtual Service	67
2.2. Virtual Service Properties	67



2.3. Real Server Assignment	69
2.4. Add / Modify Real Server	69
2.5. Add Rule	70
2.6. Rule Precedence.....	70
3. Global Settings	70
3.1. Content Rule Management.....	70
3.2. Service Health Check Parameters	70
3.2.1. Check Interval	70
3.3. Connect & Response timeouts	71
3.4. Re-try Count	71
4. Adaptive Parameters	71
4.1. Adaptive Interval	71
4.2. Adaptive URL.....	71
4.3. Port.....	71
4.4. Min Control Variable Value.....	71
4.5. Min Weight Adjustment Value.....	71
4.6. Real Server Availability	71
5. Balancer Metrics	72
5.1. Global Metrics	72
5.1.2. Real Server Metrics.....	72
5.1.3. Virtual Service Metrics.....	72
6. System Properties	72
6.1. Route Management.....	72
6.2. Access Control	72
6.2.1. Packet Filter Enabled.....	72
6.2.2. Reject/Drop blocked packets	73
6.2.3. Access control Lists.....	73
6.2.4. Add Address	73
6.3. Miscellaneous	73
6.3.1. SNAT Control	73
6.3.2. Set Transfer Protocol.....	73
6.3.3. Set HA Timeout	73



Section I. Application Guide

A. Preface

1. Foreword

Thank you for purchasing KEMP's LoadMaster!
We wish you much success with your KEMP's LoadMaster.

2. LoadMaster vs. SSL-Master Feature/Function Comparison

Feature/Function	LoadMaster 3620	SSL-Master 1020
Load Balancing (scheduling) Algorithms Supported	Round Robin Weighted Round Robin Least Connections Weighted Least Connections Fixed Weighting Adaptive	Round Robin ONLY
Layer 7 Content Switching	YES	Not Available
Persistence (stickiness)	Source IP address SSL Session ID URL Host Header Passive Cookie Active Cookie (Insert) Cookie Hash Cookie Hash Source Query Hash	Source IP ONLY

NOTE:

The SSL-Master 1020 can be upgraded to full LoadMaster 3620 functionality via a license update. To purchase the LoadMaster Full Load Balancing upgrade license please contact KEMP Technologies.

3. Typographical Conventions

Screenshots and photographs may be design models and might not correspond exactly to real-life components.

4. Glossary and Abbreviations

Access Code: An Access Code will be generated during the initial setup of the LoadMaster. You must contact your KEMP Technologies representative for your 60-day evaluation or your full purchased license key.

Balancer: A network device or logic that distributes inbound connections with a common source address across a farm of server machines.



Farm Side: The LoadMaster network interface to which the server farm is connected.

Flat-based: The VIPs and the real servers are on the same subnet.

HA: Highly Available or High Availability (used interchangeably)

ICMP: Internet Control Message Protocol

MIB: Management Information Base, a database of object definitions. The definition specifies whether an SNMP manager can monitor the object.

NAT: Network Address Translation

NAT-based: The VIPs and the real servers are on different subnets.

Network Side: The LoadMaster network interface over which requests to the server farm are made.

One-armed: Only one Ethernet interface is used for in and outbound traffic. Farm side and Network side are both connected to it.

RS: Real Server: Physical server machines which make up a server farm.

Service: A Service is an application that is connected to the network.

Shared IP: The shared (floating) IP address is always the assigned IP address of the active LoadMaster in a HA solution.

SCP: Secure copy command of SSH

SNMP: Simple Network Management Protocol, a network protocol used to manage TCP/IP networks. This protocol provides functions that enable you to access the data object whose definitions are located in the MIB.

S-NAT: Network Address Translation for a source IP address.

SSH: Secure Shell Protocol

Two-armed: Two Ethernet interfaces are used for in and outbound traffic, one connected to the network side and one to the farm side.

UTC: Universal Time Coordinated

VIP: Virtual IP Address: The IP address of a service defined on the LoadMaster.

VS: Virtual Service: An entry on the LoadMaster over which a service being hosted in the server farm can be reached.

WUI: Web User Interface used to perform LoadMaster administration via a web browser.



B. Overview of the LoadMaster

1. Load Balancing and its Benefits

As Internet-offered services and applications evolve and become more sophisticated, they also become more complicated and unwieldy for those who operate them. Consequently, the following issues become key in providing users with the Quality of Service they expect and require:

- Scalability of server machines – as the demand on services and applications grows, it is no longer sufficient to simply upgrade the hardware that hosts these units. Sooner or later, physical limits of hardware upgradeability are reached. Furthermore, users are not willing to accept the downtime that accompanies such upgrades.
- High availability of services and applications – as Internet based networking is used for mission critical applications such as banking, B2B and Voice over IP, the availability of services can easily determine the success or failure of a business.
- Greater flexibility – as the number and diversity of Internet services and applications increases, it becomes imperative for network managers to have an environment in which they can trivially “juggle” their resources on demand, without endangering the robustness of the environment.
- Improved performance – mission critical services and applications require deterministic response times. The competition is only a click away.

The Solution . . .

Load Balancers are highly robust network devices, and have the effect of making multiple server machines appear as one. Thus, a network service can be distributed across an array of physical machines – sometimes referred to as a server or application farm. The Load Balancer channels requests to a network service using a variety of intelligent scheduling methods. This has several benefits, which is why load balancing has become a successful and widely employed technique in recent years:

- Scalability of server machines – can be achieved by simply adding server machines to the farm as demand for services increases. Upfront investments in server capacity, which may go unused, can be avoided.
- High availability of services and applications – since services are replicated across multiple machines within the farm, the loss of a single server does not result in a total loss of service for the customer – worst case, a drop in performance may be experienced.
- Greater flexibility – server machines can be added and removed from the farm on demand resulting in an immediate effect. Services – sometimes referred to as Virtual Services in the context of Load Balancers – can be trivially tuned and shuffled around the server farm, thus ensuring optimal usage of resources.
- Improved performance – intelligent load balancing algorithms ensure that requests are directed to those machines in the server farm that can most effectively handle them.

The LoadMaster Load Balancer provides a more reliable, flexible and cost effective solution to addressing high traffic load and mission critical applications, resulting in a high Quality of Service (QoS) at a low Total Cost of Ownership (TCO).

2. Considerations in Getting Started

Skip this section if you are already familiar with the basic Load Balancer functions and know how to set up a Balancer-Server combination.



The Installation and Configuration Guide describes how to go about installing and configuring your LoadMaster such that load-balanced services (virtual services) can be supported. However, before doing so, the following considerations covered by the LoadMaster documentation should be taken into account when setting up your LoadMaster for the first time:

What sort of LoadMaster network topology best suits my application? [See section C of this guide]

Do my real servers require publicly routable IP addresses or can they be “hidden” behind the LoadMaster on a private network segment? [See section C of this guide]

Does my Application require the Balancer and Real Servers to be part of a flat-based topology? [See section C of this guide]

Will network connections be initiated from within the application farm to the outside, as well as from the outside to within the farm? [See section D of this guide]

Do I require a High Availability support in the form of an active/hot-standby redundant cluster? [See section E of this guide]

How do I intend to replicate my application across multiple real server machines?

What kind of Virtual Service settings best suits my application? [See section F of this guide]

Will a round robin balancing algorithm do or do I need to take aspects of my application into consideration and report these back to the Balancer? [See section F of this guide]

Is connection persistency at all an issue for my application? [See section G and H of this guide]

Will source-based IP persistency suffice or do I need to take layer 7 aspects into consideration? [See section G and H of this guide]

Do I wish to integrate the LoadMaster into my current SNMP environment? [See section J of this guide]

Which forms of real server health checking will best suite my application? [See section K of this guide]

Do I prefer a Command Line Interface for provisioning my Virtual Services or do I require the use of the Web based interface? [Refer to the Command Line Reference Guide Section III, and the WUI Handbook]

Do I wish the LoadMaster event logs to be reported to a central Syslogd? [See section N of this guide]

Do I require remote access to the CLI? [See section E of the Installation and Configuration Guide]

Do I wish to allow my balancer to be accessed by KEMP Technologies for maintenance purposes? [See section E of the Installation and Configuration Guide]



3. A Simple Balancer Configuration

Taking the above issues into consideration, an example Load Balanced Site may look as follows:

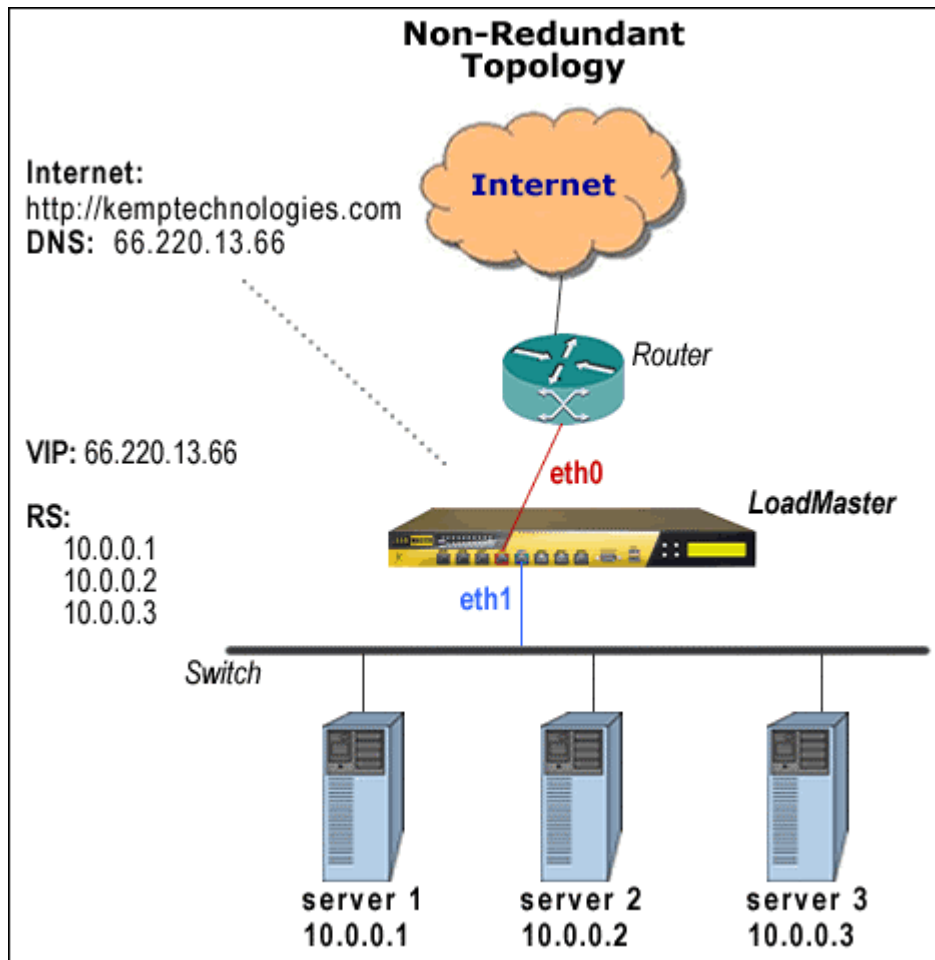


Figure 1: Example of a simple Balancer configuration

A Virtual Service (VS) has been created on the LoadMaster with IP address 66.220.13.66 for an http service.

This Virtual Service has been configured to balance the incoming traffic across real servers (RS) server 1, server 2 and server 3.

A client requests "http://www.kemptechnologies.com".

The URL will be resolved into 66.220.13.66.

The request will be routed to the LoadMaster, which offers this IP address as an IP-alias of its network interface eth0.

The LoadMaster is connected to the server farm subnet 10.0.0.0 via its network interface eth1.

The LoadMaster knows that in this subnet were three Real Servers assigned to the requested address 66.220.13.66 and able to deliver the required content.



The LoadMaster uses the load balancing method you configured - e.g. weighted round robin - and distributes the request onto one of the three Real Servers.

4. LoadMaster Load Balancer Features

The LoadMaster load balancer provides the following features with the Balancer Operating Software and the Web User Interface:

The Balancer Operating Software Basics

- Server Load Balancing (SLB) for TCP/UDP based protocols
- Multi-armed Balancer/Server Farm network topology with NAT-based forwarding. See note 1.
- Compact Flash bootable Operating Software
- S-NAT support for multi-armed solutions. See note 2.
- One and two-armed flat-based 3 Balancer/Server Farm topologies
- Support of Direct Server Return (DSR) configurations
- Optional force setting of duplex mode for Balancer Ethernet interfaces
- Option to allow remote access to Balancer for maintenance purposes
- VLAN tagging support

NOTES:

1. Real Servers and the Virtual Services are maintained in different logical networks using a NAT-like forwarding mechanism. For this purpose the LoadMaster uses two Ethernet interfaces (hence two armed).
2. If S-NAT is enabled, then connections initiated from a Real Server to the Internet assume the source IP address of the balancer.

Scheduling and L4 / L7 Persistency

- Four static load-balancing methods
- Agent based automatic adaptive balancing with real server API
- Connection persistency based on:
 - Source IP address
 - SSL Session ID
 - URL
 - Host Header
 - Passive Cookie
 - Active Cookie (Insert)
 - Cookie Hash
 - Cookie Hash Source
 - Query Hash
- Port following for persistency options
- SSL acceleration



Health Check and Availability

- ICMP health checking of server farm machines
- Service checking for DNS, FTP, HTTP, HTTPS, IMAP, NNTP, POP3, SMTP, TELNET
- Automatic reconfiguration of defective real server machines
- Active/Hot-Standby configurations for High Availability as an option
- Stateful Failover of Cookies and TCP connections

Administration

- Web based interface for creation, deletion and editing of Virtual Services
- Command Line Interface (CLI) for the creation, deletion and editing of Virtual Services
- Packet filtering functionality
- Remote Syslogd support
- Remote access to the LoadMaster for all administrative Balancer operations
- Selective restore of Balancer and Virtual Service data
- Online software upgrades for LoadMaster Operating Software
- SNMP support for event traps
- SNMP for performance metrics

Miscellaneous

- Support of Time Zones
- Change password function for administrative login "bal"
- Password recovery mechanism for login "bal"
- Multi-language keyboard support

C. LoadMaster Network Topologies

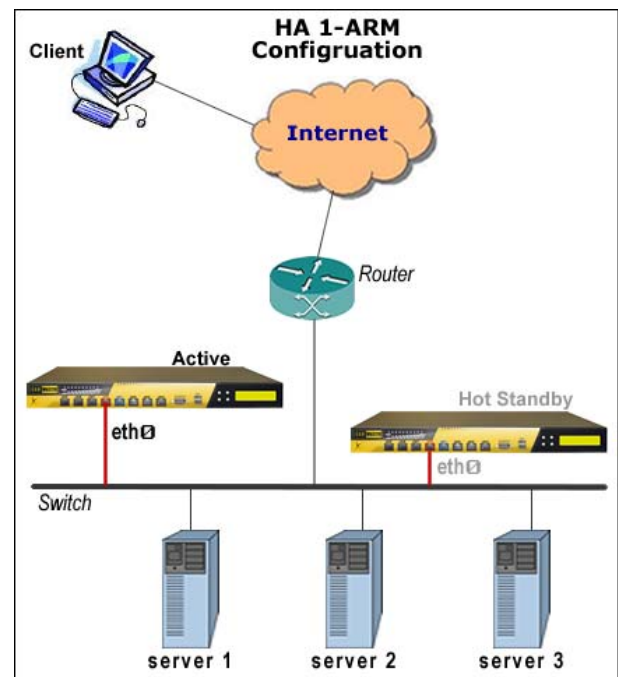
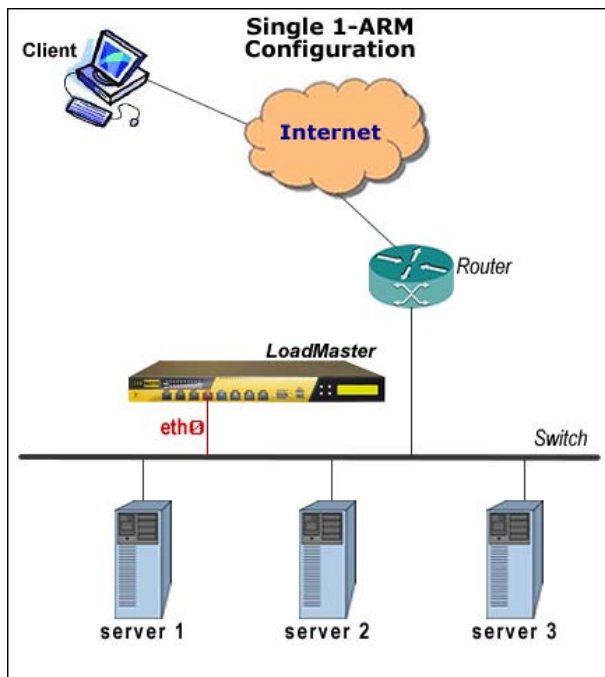
1. One-Armed Balancer

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

- Only the eth0 Ethernet interface will be used (for both in and outbound traffic)
- 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.
- S-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



- Implies the clients (consumers of the service hosted by the LoadMaster) are on a logically separate network to the LoadMaster and its Virtual Services (this is not true if used in conjunction with DSR).

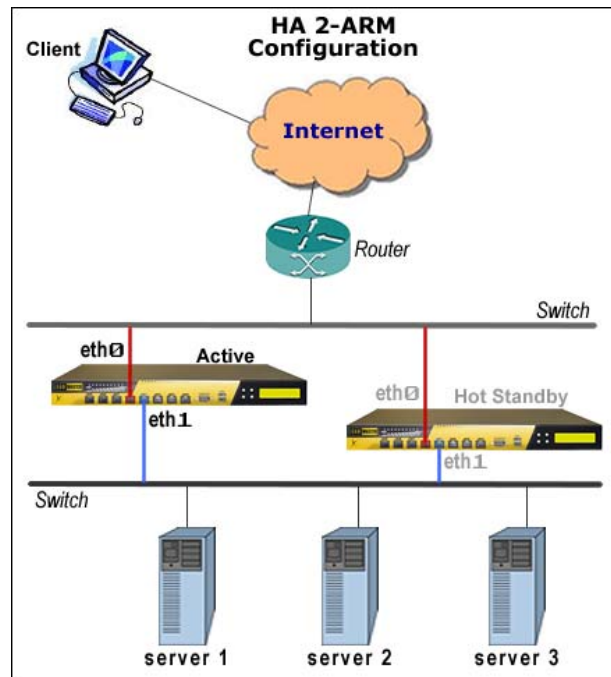
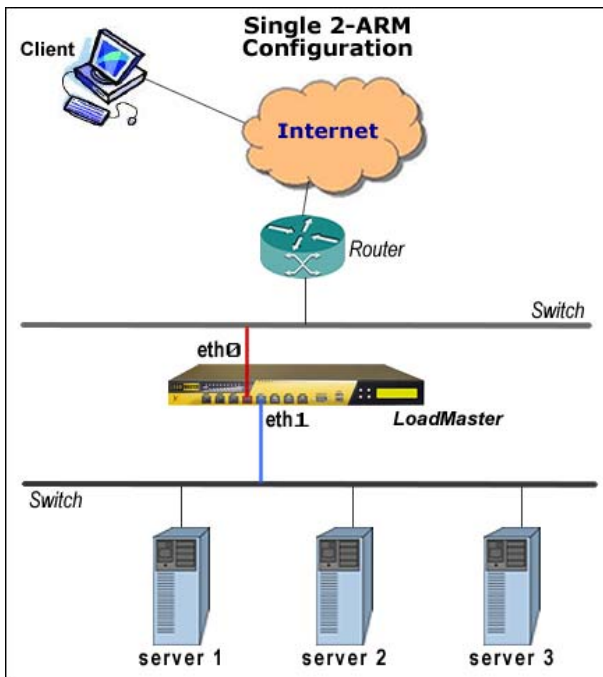


2. Two-Armed and Multi-Armed Balancer

If a two-armed or Multi-Armed configuration is selected, then the following is true:

- 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 IP addresses
- S-NAT may be useful in such a configuration
- Clients may be on the same logical network as the LoadMaster
- Virtual Services may be created on either eth0 or eth1. Up to eth7 on Multi-Armed configurations
- Real Servers may exist on either the eth0 or up to the eth7 network. However, placing Real Server on eth0 in a two-armed configuration is not recommended.





3. Direct Server Return – DSR example

- 1 – incoming request intercepted by LoadMaster
- 2 – routed to Server 1
- 3 – response from Server 1
- 4 – Response goes directly to Client without LoadMaster

Figure above: Sample Direct Server Return configuration

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 should have a path to the clients without going through the balancer, i.e. additional routes bypassing the LoadMaster.

Note: This mode is only available when not using any persistency options.

DSR uses a combination of MAT (MAC address translation) and a special RS 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 the Real Server should be configured so that they do not respond to arp requests. For Linux with a recent 2.4 kernel, this can be done by creating the VIP as an IP alias on the loopback interface.

When you create the VS and assign the respective real servers to it, select "route" as the forwarding method to the real servers. This means that the balancer just routes the packets from a client to a RS without modifying the IP addresses. The real server accepts requests for the VIP destination address because it has configured the VIP as an IP alias. The real server will then reply to the IP address of the requesting client with the source IP address of the reply set to the VIP.

Step	Source IP	Destination IP	MAC Address
1	216.139.43.10	195.30.70.200	Dest: 00:00:00:00:00:aa
2	216.139.43.10	195.30.70.200	Dest: 00:00:00:00:00:bb
3	195.30.70.200	216.139.43.10	Source: 00:00:00:00:00:bb



Configuring a VIP on the loopback interface on Linux

On a linux machine the “ifconfig –a” command will look something like this:

```
root@RS1 $ ifconfig -a
eth0  Link encap:Ethernet HWaddr 00:00:00:00:00:bb inet addr: 195.30.70.11 Bcast: 195.30.70.255
      Mask: 255.255.255.0 UP BROADCAST RUNNING MULTICAST MTU: 1500 Metric: 1 RX packets: 96561817
      errors: 526 dropped: 0 overruns: 5 frame: 0 TX packets: 97174301 errors: 0 dropped: 0 overruns: 0
      carrier: 0 collisions: 0 txqueuelen: 100 Interrupt: 10 Base address: 0x4000
lo    Link encap:Local Loopback inet addr: 127.0.0.1 Mask: 255.0.0.0 UP LOOPBACK RUNNING MTU: 3924
      Metric: 1 RX packets: 3985923 errors: 0 dropped: 0 overruns: 0 frame: 0 TX packets: 3985923 errors: 0
      dropped: 0 overruns: 0 carrier: 0 collisions: 0 txqueuelen: 0
```

To create an additional loopback interface with an IP alias use the “ifconfig” command like this:

```
root@RS1 $ ifconfig lo:1 195.30.70.200 broadcast 195.30.70.200 \ netmask 255.255.255.255
root@RS1 $ ifconfig lo:1
lo:1  Link encap:Local Loopback inet addr: 195.30.70.200 Mask: 255.255.255.255 UP LOOPBACK RUNNING
      MTU: 3924 Metric: 1
```

D. Miscellaneous Networking Issues

1. S-NAT

When using a two-armed or multi-armed balancer configuration, it is sometimes useful for the real servers to have access to the Internet. The default route for the real servers is through the balancer. If however the real servers do not have routable addresses i.e. private addresses, this is not possible.

Using S-NAT, the balancer will map all connections originating on a real server so that they appear to come from the balancer itself. The real servers can thus use the Internet as if directly connected but with the extra security protection that they cannot be addressed directly from the Internet.

The use of S-NAT in single-armed configurations is not recommended.

The S-NAT functionality may be enabled or disabled via the configuration menus and WUI.

2. Default Gateway and Routes

In simple configurations, where the LoadMaster is installed in a network where there is only one route to the Internet, only the default gateway needs be specified. All traffic from the LoadMaster to the Internet will then be routed over this gateway. An example configuration is given in figure A.

When the LoadMaster is installed in a more complicated network configuration (for example as depicted in figure B), the default gateway must still be specified but will only be used if additional routes are not available. Additional routes may be specified so that traffic for the specified addresses will be routed over alternative gateways. For example in figure B, a route could be set up to route data from a private network or over a secondary link gateway.

Only static routes can be set up on the LoadMaster (see the Installation and Configuration Guide in this handbook). The balancer does not currently support external routing protocols.



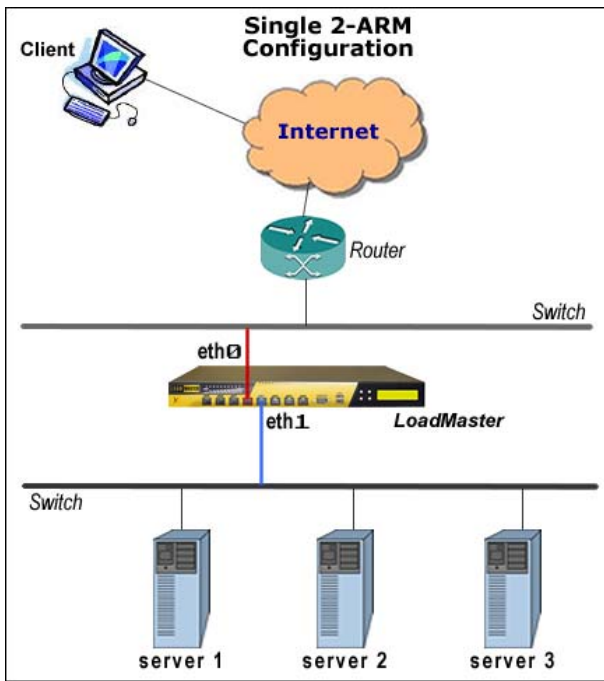


Figure A.

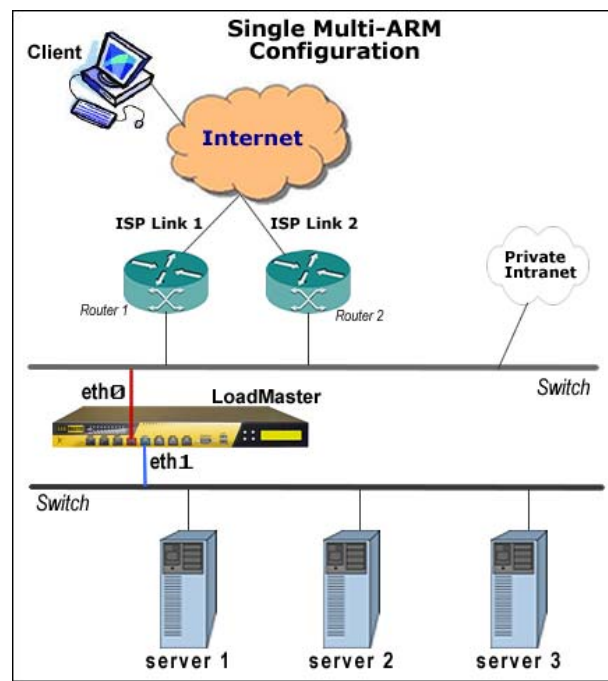


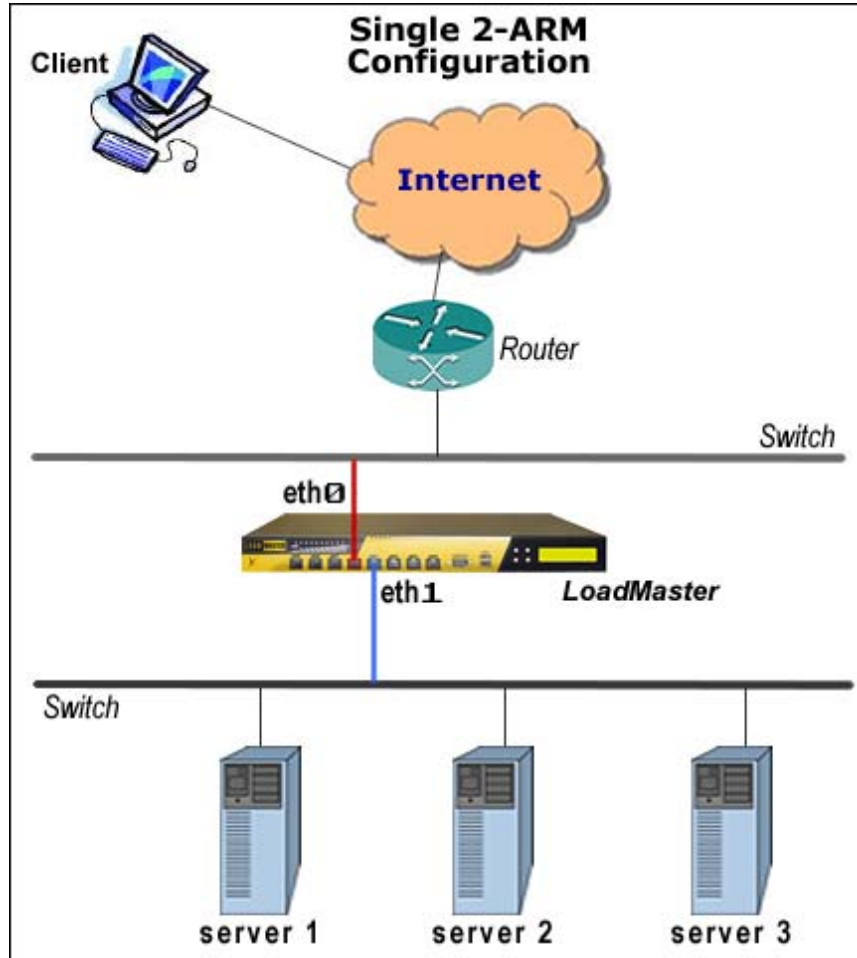
Figure B.



E. Single/Dual Unit Configurations

1. Single Unit Configuration

The topology in standalone-mode looks like this:



2. High Availability (HA) Configuration

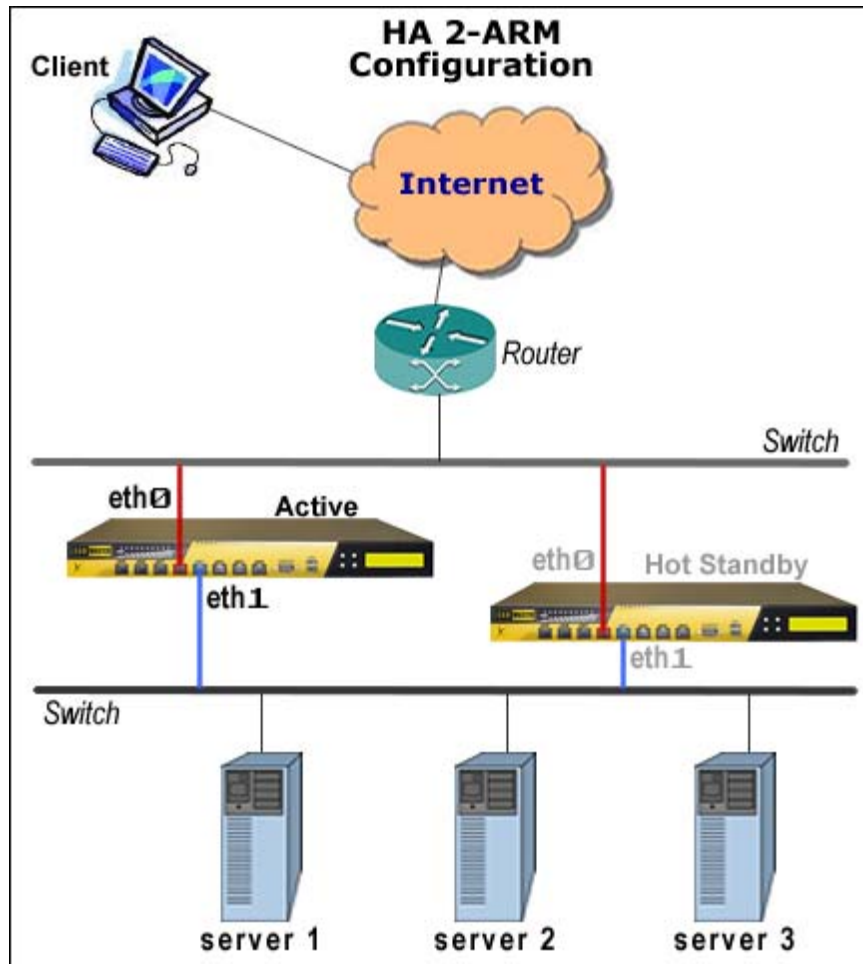
The High Availability feature of the LoadMaster guarantees the availability of your server farm. HA is achieved by a hot-standby, failover mechanism. Two identical LoadMaster units are integrated into your network. One machine serves as the active balancer, the second one remains in a standby, idle state,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

Note: If you are running a High Availability (HA) cluster, which consists of two interconnected LoadMaster nodes, each network interface has an individual IP address and a shared (sometimes called floating) IP address. The shared IP address is identical for both LoadMaster nodes, but active only for one LoadMaster at any one time.

During normal operation each node is periodically sending health check messages over the two connections to cross check the operability of the peer machine. In the unlikely event that a LoadMaster should fail, the standby machine will become active and take over the task of balancing.



The topology in high availability-mode looks like this:



Note: In HA mode, the Real Servers must have the shared IP address of the LoadMaster farm-side interface configured as the default gateway.

F. Balancing Methods

There are several load balancing methods provided by the LoadMaster, which are known as "scheduling rules" or "algorithms":

1. Round Robin

With this method incoming requests are distributed sequentially across the server farm (cluster), i.e. 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.



2. 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.

3. 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.

4. 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.

5. Agent Based Adaptive Balancing

In addition to the methods above the LoadMaster contains an adaptive logic, which checks the state of the servers at regular intervals and independently of the configured weighting.

For the extremely powerful "agent based adaptive balancing" method the balancer periodically checks the system load on all the servers in the farm: Each server machine should provide a file that contains a numeric value in the range between 0 and 100 representing the actual load on this server (0 = idle, 100 = overload). The balancer retrieves this file by an HTTP GET operation. It is the server's job to provide the actual load in the ASCII file. There are no prerequisites how the servers evaluate this information.

Two different strategies are applied, depending on the overall load of the server farm: During normal operation 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. As with the weighted round robin, incorrect distribution can then be countered by assigning different weights to the servers available.

During a period of very low traffic, however, the load values as reported by the servers will not build a representative sample. A load distribution based on these values would result in uncontrolled, oscillating directives. Therefore in such a situation it is more reasonable, to calculate the load distribution based on the static weight ratio. The balancer switches to the weighted round robin method automatically when the load on all servers falls below a limit defined by the administrator. If the load rises above the limit the balancer switches back to the adaptive method.



6. Fixed Weighting

Fixed scheduling means that the Real Servers are ranked via their weights. The Real Server with the highest weight (which is not down) will handle all the requests. This is useful if there are two servers, the main server and a backup server. The main server handles all traffic until it has a failure, the backup server will then handle the traffic only until the main server is again available, at which time, traffic will switch back to the main server.

(See also section O.1: API for Agent Based Adaptive Balancing)

G. Layer 4 Persistency

1. Source IP Address Based Persistency

The LoadMaster can balance TCP or UDP based traffic based on source and destination IP addresses. All packets are passed through to one of the real servers. It can also ensure that requests from one particular host always go to the same server. This is known as connection persistency.

Persistency in the context of load balancing means that multiple requests from a client to a virtual service are redirected to the same real server as that selected for the first request.

The duration period for which the LoadMaster maintains the persistency for a given source IP address can be controlled via the persistency timeout. The persistency timeout is specified in minutes; otherwise the default value of 6 minutes (360 seconds) will be used.

It is also possible to specify an optional persistency mask that determines the granularity of the source IP match. For a persistency mask of 255.255.255.255 (the default) every single source IP address is considered separately, whereas with a persistency mask of 255.255.255.0 only the leading 24 bits of the source IP address are considered for the persistency i.e. any source IP address from this network is directed to the same real server.

This feature can be used when clients are connected via Proxies, where the client source IP seen by the balancer may vary between requests. The thought here is that the addresses used by the proxy fall in contiguous ranges or blocks (e.g. class c net) that can be masked with the persistency mask.

In some situations, most notably when a device that performs full NAT is placed in front of the LoadMaster, all requests arrive at the LoadMaster from only one IP address. This totally defeats IP source address based persistency, all requests would go to only one real server, while the rest of the server farm idles. This may be a reason to employ the Layer 7 Persistency options.

H. Layer 7 Persistency

The LoadMaster supports several content-based persistency methods. These are normally termed layer 7 persistency methods because they use the content of the message to determine which real server should be used. When using these methods, the LoadMaster intercepts a request and looks at the start of the message from the client, using this information, the LoadMaster can determine to which real server the connection should be routed.

These methods make most sense for protocols such as HTTP and SSL and are therefore explicitly supported by the LoadMaster.

1. SSL session ID Based Persistency

The SSL (Secure Socket Layer) protocol is used on the World Wide Web to protect confidential information, by performing authentication, data encryption and ensuring message integrity. The LoadMaster uses the SSL



session ID to ensure that all traffic for an SSL transaction reaches the same real server. This is a “common” feature for commercial, financial and shopping-cart based web sites.

Note: This mode is not available if SSL acceleration is enabled. SSL in this case would be achieved via the Real Server, which can lead to performance hits.

2. URL Based Persistency

Using this method, the LoadMaster will direct requests for the same URL to the same real server as long as the persistency duration is valid.

3. URL Host Based Persistency

This unique feature of the LoadMaster allows a single balanced site to support multiple addresses. Each address with the same host will then be routed to the same real server as long as the persistency duration is valid. This is useful when a single site is addressed by different URLs (i.e. mysite.org and mysite.foryou.edu could both go to the same virtual address but they could be handled by different real servers).

4. Cookie Based Persistency

In this mode, the real servers supply a server specific cookie, which the LoadMaster uses to determine to which real server the next request with the same cookie should be sent. When using passive cookie handling, each real server must be modified to send a “Set-Cookie” record on each request (which does not already have a cookie value set). (For an example refer to Section O.2)

5. Cookie/Source (Cookie or IP source) Based Persistency

This method functions exactly like “cookie based persistency” when the client browser returns a cookie. If however the client does not return a cookie, either because the user has disabled them or has ignored the cookie, then the clients IP address will be used as the persistency key.

6. Active Cookie Based Persistency

In this mode, the LoadMaster inserts a cookie into the data stream returned to the client when a request is made. This cookie is used to identify which real server handled the request. When the client again makes a request, the cookie will be sent back to the LoadMaster, which will then use the value of the cookie to direct the request to the same real server as before. When using active mode, no modifications must be done to the real servers. The Balancer administers all cookies for the servers.

7. Active Cookie/Source (Active Cookie or IP source) Based Persistency

This method functions exactly like “active cookie based persistency” when the client browser returns a cookie. If however the client does not return a cookie, either because the user has disabled them or has ignored the cookie, then the clients IP address will be used as the persistency key.

8. Cookie hash Persistency

This method “hashes” the value of all cookies sent by the client to determine which real server to direct the request to. All requests that have the same set of cookies will be sent to the same real server.

9. 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.



I.e. 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.

Note: This only works correctly if both services have the same set of real servers.

1. SSL Acceleration

When this option is enabled. The B-100 functions as an SSL endpoint and decrypts the content of the message. This allows the B-100 to use the contents of the message to perform content switching and handle persistency options. In this mode, the connection to the real server is not encrypted. This relieves the real server of the work needed to handle the SSL protocol and also means that only one SSL certificate is required for the B-100, instead of one certificate per real server. SSL acceleration is available for any "tcp" service if the persistency option is not SSL. The use of SSL acceleration and SSL persistency does not make sense and so is not supported.

1. Reverse Acceleration

When this mode is enabled, the LoadMaster functions as an SSL proxy client and encrypts any input that it receives before sending it to the real server. Only Source based persistency is permitted in this mode.

If a service is to have reverse acceleration, no certificate is required and the user can ignore the request to install one. It is the responsibility of the user to install a suitable certificate on the real server.

This mode can be used in conjunction with a second virtual service to provide end-to-end SSL encryption.

Warning:

Connecting the output of one VS to the input of a second VS only works if L7 transparency is turned OFF. This means that the LoadMaster cannot be transparent when using end-to-end encryption.

2. Certificate Files

To enable the LoadMaster to act as an SSL endpoint, a server SSL certificate and a private key must be supplied. When SSL acceleration is enabled via the Web interface, the user is given the option of installing a SSL certificate file. It is also possible to install and update certificates via the administration menu. If no certificate is installed for a virtual service, a temporary one is generated locally on the LoadMaster. This certificate is locally signed and will be regenerated upon reboot and so should not be used except for testing. An SSL certificate can be acquired from one of the various Trusted Authorities.

A private key must also be transferred to the LoadMaster. This can be either a separate file on the same machine or the private key can be appended to the SSL certificate. If the private key is appended to the SSL certificate (openssl can generate one file with both parts in), then a separate private key is not required. Note: A Certificate file which also contains a private key or a private key file must be installed first. The certificate and the private key are checked to see that they are compatible. If they are not, then the certificate file will not be installed.

Using the Web interface, only PEM encoded certificates and keys may be installed. When using the administration menu, certificates which are encoded in PEM or in binary or which are packed in certificate tree files can be installed.



3. 3rd Party Certificates

The scheme described above works well for certificates which are fully trusted, these types of certificates are expensive and are only available from certain Trusted Authorities. It is now common practice for Trusted Authorities to issue certificates which depend on a chain of other more trusted certificates, which are supplied by a Trust Authority. To enable the LoadMaster to properly use the supplied certificates, the certificate chain must also be installed on the LoadMaster. This is done using the administrative menu.

J. Rule Based Content Switching

In the previously described load balancing methods, it is assumed that all real servers have the same content. A network administrator may wish to split up the site so that certain servers should be used only for static content i.e. ".gif"s while other servers should be used only for scripts etc. This would be useful if a customer database only ran on a small number of servers but the site had lots of static content.

With L7 content switching, "Rules" can be defined which can be used to redirect different requests to different real servers. A "rule" defines a string, which is matched against the incoming URL string. If the URL matches the "Rule", the request is directed to the Real Servers the "rule" is assigned to.

URL definition according to RFC2616:

`http://www.a-host.com/content/example/request.cgi?value="hello"`
|← →|← →|← Start URL End URL →|← Query part →|
Protocol definition and Host part (ignored) (ignored)

1. Rule Definition

A "rule", can be defined as one of the following:

A prefix string: The rule will match if the start of the URL matches the given string.
I.e. If a rule has the value "/home", then all requests for "/home/..." will be matched by the rule.

A postfix string: The rule will match if the end of the URL matches the given string.
I.e. If a rule has the value ".gif", then all requests for "XXX.gif" will be matched by the rule.

A regular expression: The rule will match if the URL contains a string, which matches the given Regular expression.
I.e. If a rule has a value "home/*.gif", then all requests for "/home/.../XXX.gif" will match the rule.

Any rule can also be negated: In this case, the rule will match if the string does NOT match the given rule.
I.e. if a NEGATED postfix rule has the value ".gif", then it will match all requests which do NOT end in ".gif". For example the URL "/home/.../XXX.jpg" will be matched by the rule.

1.1. Special Characters

The strings used for the prefix and postfix operations can contain any character, which is permitted in a URL. "Special" Regular expression characters are taken as literal characters. I.e. When using prefix or postfix operations a rule such as "home/*" will only match if the string contains a "*" character at the relevant position.

The use of the "%<hex><hex>" characters is allowed in all strings. This construct will allow a user to input "Unsafe" characters into the string. The construct "%%" will be treated as a single "%" during matching.

All match strings are case dependent.



FTP	21	TCP
TELNET	23	TCP
SMTP	25	TCP
HTTP	80	TCP
HTTPS	443	TCP
POP3	110	TCP
NNTP	119	TCP
IMAP	143	TCP
DNS	53	UDP

For other ports the LoadMaster uses Layer4 health checks for TCP services and Layer3 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 Layer4 check.

Note: These global settings hold for all servers in the farm, i.e. 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.

1. Service and non-service based Health Checking

Layer3 health checks utilize ICMP based echo requests (pings) to test whether a real server can be reached over the network. A Layer3 check is not Virtual Service specific, e.g. when it fails, the corresponding Real Server will be removed from all Virtual Services that use it.

Service Based Health Checking In contrast to the Layer3 health checks, both the Layer4 and Layer7 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.

<u>Layer</u>	<u>Type</u>	<u>Description</u>
3	ICMP	The LoadMaster sends ICMP echo requests (pings) to the Real Servers. A Real Server fails this check when it doesn't respond with an ICMP echo response in the configured response time for the configured number of retries.
4	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.
7	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.
7	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 'Oxff', 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, it is assumed dead.
7	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.

- | | | |
|---|-------|---|
| 7 | 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) 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. |
| 7 | HTTPS | The LoadMaster opens a 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) 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. |
| 7 | POP3 | The LoadMaster opens a TCP connection to the Real Server on the Service port (port 110). If the server respond with a greeting message that start 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 different status code, it is assumed dead. |
| 7 | 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. |
| 7 | IMAP | The LoadMaster opens a TCP connection to the Real Server on the Service port (port 143). If the server respond with a greeting message that start 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. |
| 7 | DNS | The LoadMaster sends Source-of-Authority (SOA) request to the Real Server on the Service port (port 53 UDP). If the server successfully responds to the SOA request, 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 SOA request, it is assumed dead. |

Refer to Section III Command Line Interface Reference Guide, on how to best configure your balancer for health checking.

L. SNMP Support

Simple Network Management Protocol (SNMP) is a protocol that allows one to manage 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, e.g. such as a unit failover. The alerting mechanism uses so-called event traps.



For a description of the SNMP standard see **References** (p. 37). The current version is SNMPv3, the two other major revisions in use are SNMPv1 and SNMPv2c (community-based SNMPv2).

The SNMP support of the LoadMaster is based on SNMPv3, so that all 3 of the above versions can be used. However, since SNMPv1 does not support 64bit-values (as used in the LoadMaster MIB), it is recommended to use SNMPv2c or SNMPv3.

1. LoadMaster Performance Metrics via SNMP

The information regarding all LoadMaster-specific data objects is stored in three enterprise-specific MIBs (Management Information Base).

ONE4NET-MIB.txt	<i>enterprise id</i>
IPVS-MIB.txt	<i>Virtual Server stats</i>
B-100-MIB.txt	<i>LoadMaster configuration data</i>

These MIBs (located on the LoadMaster CD) need to be installed on the SNMP manager machine in order to be able to request the performance-/config-data of the LoadMaster via SNMP.

The description of the counters can be taken from the LoadMaster MIBs (the description clause).

Apart from just reading the MIB this can be done for Linux (nag ucdsnmp) with the command:

```
snmptranslate -Td -OS <oid>
```

where <oid> is the object identifier in question.

Example: <oid> = .1.3.6.1.4.1.one4net.ipvs.ipvsRSTable.rsEntry.RSConns

```
snmptranslate -Td -Ov .1.3.6.1.4.1.one4net.ipvs.ipvsRSTable.rsEntry.RSConns
```

```
.1.3.6.1.4.1.12196.12.2.1.12
RSConns OBJECT-TYPE
-- FROM          IPVS-MIB
SYNTAX           Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "the total number of connections for this RS"
 ::= { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1)
       one4net(12196) ipvs(12)
       ipvsRSTable(2) rsEntry(1) 12 }
```

The data object defined in the LoadMaster MIBS is a superset to the counters displayed by the WUI.

Note: The data objects on the LoadMaster are not writable, so that only GET requests (GET, GET-NEXT, GET-BULK,...) should be used.

For a description of the configuration of SNMP support on the LoadMaster refer to the installation guide. The SNMP support is disabled by default.

2. LoadMaster Event Traps via SNMP

The LoadMaster supports the generation of SNMPv1 and SNMPv2 traps.

When the feature is enabled, the following traps are generated:

ColdStart	generic (start/stop of SNMP sub-system)
VsStateChange	(Virtual Service state change)
RsStateChange	(Real Server state change)
HaStateChange	(HA configuration only: LoadMaster failover)



For the configuration of trap feature refer to the installation guide.

Trap generation is disabled by default.

References

SNMPv1:

- RFC 1155 Structure and Identification of Management Information for TCP/IP-based Internets
- RFC 1157 A Simple Network Management Protocol (SNMP)
- RFC 1212 Concise MIB Definitions

SNMPv2c:

- RFC 1901 Introduction to Community-based SNMP
- RFC 1902 Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)
- RFC 1903 Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)
- RFC 1904 Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)
- RFC 1905 Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)
- RFC 1906 Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)
- RFC 1907 Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)

SNMPv3:

- RFC 2570 Introduction to Version 3 of the Internet-standard Network Management Framework
- RFC 2571 An Architecture for Describing SNMP Management Frameworks
- RFC 2572 Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
- RFC 2573 SNMP Applications
- RFC 2574 User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
- RFC 2575 View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)
- RFC 2576 Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework

M. LoadMaster Software Upgrades

1. Online Upgrades

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

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



Using the configuration menu (utilities->software upgrade), it is possible to download the patch from the server machine (the protocol used can be FTP, SCP or HTTP). 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.

If for some reason, the patch does not perform as required, the previous version of the software may be reactivated via the configuration menu.

Converting from a 30 or 60 day evaluation license to a full license or from a L4 only to a L4 and L7 license can be performed using the menu item Utilities->Update License. If the LoadMaster is already running in a L4+L7 mode, this menu option is not available.

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

N. Miscellaneous

1. Remote Syslogd Support

The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally and may be displayed via the diagnostics menu point. It is also possible to configure the LoadMaster to transmit these error messages to a remote syslogd server (menu point: extended->syslog). Five different error message levels are defined. Each level of message may be sent to a different server. The levels are:

NOTICE
WARN
ERROR
CRITICAL
EMERGENCY

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

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

2. How to get a license

After boot, a login prompt appears; login as 'bal' (password '1fourall').

To unlock the LoadMaster software you need a license key. The license key will be generated individually for each single LoadMaster instance in conjunction with a hardware dependent Access Code.

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

- a) An evaluation license. This is a fully functional license valid for up to 30 days.
- b) A full, non time-limited LoadMaster license.
- c) A full, non time-limited license for a LoadMaster High Availability (HA) cluster consisting of two machines.

An evaluation license can be upgraded to either a full single or a full HA license.

2.1. Get a 30 day evaluation license



1. Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM+ port (COM+ settings should 115200,8,N,1). After boot, a login prompt appears; login as 'bal' (password '1fourall'), and your Access Code will be displayed on the screen.
2. If not already provided contact your KEMP Technologies Representative to obtain evaluation license. Be sure to provide the Access Code (or codes if HA), so the evaluation license can be "mapped" to the unit(s). Customer contact should have provided KEMP with a valid email address to send license to Customer contact.

2.2. Get a full LoadMaster license

1. A service agreement upon purchase must be approved by KEMP in order to obtain a full LoadMaster appliance.
2. Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM+ port (COM+ settings should 115200,8,N,1). After boot, a login prompt appears; login as 'bal' (password '1fourall'), and your Access Code will be displayed on the screen.
3. If not already provided contact your KEMP Technologies Representative to obtain evaluation license. Be sure to provide the Access Code (or codes if HA), so the evaluation license can be "mapped" to the unit(s). Customer contact should have provided KEMP with a valid email address to send license to Customer contact.

2.3. Get full High Availability LoadMaster cluster licenses

1. A service agreement upon purchase must be approved by KEMP in order to obtain a full LoadMaster HA license.
2. Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM+ port (COM+ settings should 115200,8,N,1). After boot, a login prompt appears; login as 'bal' (password '1fourall'), and your Access Code will be displayed on the screen.
3. If not already provided contact your KEMP Technologies Representative to obtain evaluation license. Be sure to provide the Access Code (or codes if HA), so the evaluation license can be "mapped" to the unit(s). Customer contact should have provided KEMP with a valid email address to send license to Customer contact.

Note: TPS Limits for SSL acceleration (100 default, 400, 700, 1000) will be determined upon service agreement. Please contact your KEMP representative for more information and pricing.

Note: *The License Keys and Access Codes are NOT interchangeable between machines.*

2.4. Upgrading the evaluation license to a full single or HA license

1. A service agreement upon purchase must be approved by KEMP in order to obtain a full LoadMaster HA license.
2. Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM+ port (COM+ settings should 115200,8,N,1). After boot, a login prompt appears; login as 'bal' (password '1fourall'), and your Access Code will be displayed on the screen.
3. If not already provided contact your KEMP Technologies Representative to obtain evaluation license. Be sure to provide the Access Code (or codes if HA), so the evaluation license can be "mapped" to the unit(s). Customer contact should have provided KEMP with a valid email address to send license to Customer contact.
4. Once logged into Main Menu screen select '7' for Utilities and then '5' for Update License and enter full license key provided by your KEMP Representative. Repeat process for second LoadMaster if using HA systems.

Note: TPS Limits for SSL acceleration (100 default, 400, 700, 1000) will be determined upon service agreement. Please contact your KEMP representative for more information and pricing.



Note: The License Keys and Access Codes are NOT interchangeable between machines.

3. Backup and Restore

The configuration of a LoadMaster balancer can be saved over a network to a remote server. The complete configuration (the Virtual Service Configuration and the "base" Configuration) of the balancer will be saved to a single file on the server. The server must be running an FTP daemon or an SSH daemon. By default the remote protocol will be FTP. Using console or SSH access go to '7' Utilities, then '2' Transfer protocol to change setting. Consult the WUI User Manual to perform this function via its Web User Interface

When a configuration is restored, the user will be asked which parts of the configuration should be restored:

The Virtual Service Configuration only,

The LoadMaster "base" Configuration only,

Or the Virtual Service + the LoadMaster "base" Configuration.

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

The Virtual Service Configuration contains only the information about the Virtual Services and the real servers.

Note: When performing a restore on the standby machine of a HA cluster, only the base configuration can be restored. The Virtual Service Configuration will be taken from the active machine.

4. System recovery

A system recovery is necessary e.g. in case of a corrupted file system on the flash.

- Make sure there is a backup at hand.
- Make sure the license key is at hand.
- Contact KEMP Support to determine if recovery process can be achieved. If so then;
- Do the basic-configuration (input license key and do the quick setup)
- Restore the configuration via the backup/restore option in the Initial setup.

5. Interoperability between L4 / L7 Virtual Services

When one switches a service from one persistency method to another, the absolute values of all VS / RS counters will be reset to zero.

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

O. Appendix I

1. API for Agent Based Adaptive Balancing

For adaptive scheduling the balancer periodically checks the system load on all the servers in the farm. Each server machine has to provide a file that contains a numeric value in the range between 0 and 100 representing the actual load on this server (0 = idle, 100 = overload). The balancer retrieves this file by an HTTP GET operation. It is the servers' job to provide the actual load in the ASCII file. There are no prerequisites, though, how the servers evaluate this information.



Anyway there are some conditions that must hold:

- There must exist an ASCII file with a number between 0 and 100 in the first line.
- The file must be accessible to an HTTP GET from the balancer.
- The URL must be the same on all servers.
- The URL must match the entry "URL for adaptive scheduling" as set in the "Global Options" window.

The following is an example script to determine and present the load information on a LINUX server:

```
get load() {
  awk '/^cpu0/ {printf "%d %d %d %d\n", $2, $3, $4, $5}' \
  /proc/stat > /tmp/cpuload
  read USR SYS IOWAIT IDLE < /tmp/cpuload
  # echo $USR $SYS $IOWAIT $IDLE
}

INTV=5
DOCUMENTROOT=/usr/local/httpd/htdocs/
LOADFILE="$DOCUMENTROOT/load"

main() {
  while true; do
    USR1=$USR
    SYS1=$SYS
    IOWAIT1=$IOWAIT
    IDLE1=$IDLE
    get load
    SUM=$(( $USR+$SYS+$IOWAIT+$IDLE ))
    PUSR=$(( (USR-USR1)/$INTV ))
    PSYS=$(( (SYS-SYS1)/$INTV ))
    PIOWAIT=$(( (IOWAIT-IOWAIT1)/$INTV ))
    PIDLE=$(( (IDLE-IDLE1)/$INTV ))
    echo "$(( 100-PIDLE ))" > $LOADFILE
    sleep $INTV
  done
}
get load
main
```

Here is an example of a C program to determine and present the load information on a MS Windows NT or 2000 server:



```

#include <windows.h>
#include <stdio.h>
#include <conio.h>
#
#define CPU 0 0 3000 /*processor 0*/
#define INTERVAL MS 3000 /*three seconds as interval*/

/*counter path for Windows NT 4.0 and 2000*/
#define COUNTER_PATH NT TEXT("\\\\\\%s\\Prozessor(%d)\\%% Prozessorzeit
(
v
HQUERY hQuery
F
yo
{
TCHAR c name[MAX_COMPUTERNAME_LEN
TCHAR counter_path
DWORD ctrType;
DWORD size=MAX_COMP
HCOUNTER hCounter;
PDH_STATUS pdhStatus;

if(!GetCo
{
fprintf(stderr
.
exit

pdhStatus = PdhOpenQuery(0,0
if (pdhStatus!=NO

if((GetVersion() & 0xFF) >= 5)
sprintf
else
sp
pdhStatus=PdhAddCounter(hQue
if (pdhStatus!=NO
exit{
while{
{
fp=fopen(COU
if (!fp
{
fprintf(stderr,"ERROR: Couldn't open counter file!\n");
.
exit
}
pdhStatus=PdhCollectQueryDat
if (pdhStatus!=NO
exit(1);
pdhStatus =
PdhGetFormattedCounterValue (hCounter, PDH_FMT_DOUBLE, &ctrType, &fmt
fprintf(fp,TEXT{
fclose(fp);
.
Sle
}
}
yo
{
if (hQuery!=INVALID_HANDLE_VALUE
.

```

This example code is a program that obtains the CPU load counter from Windows 2000. It uses the Performance Data Helper (PDH) API, and must be linked to the pdh.lib.

The PDH Dynamic Link Library (DLL) pdh.dll must also be installed on the system. Modify the counter paths for Windows 2000 dependent on the installed language.

2. Http Server Configuration for Cookie Support

This short example shows how a cookie may be set on a real server.



```

#!/usr/bin/perl
#####
$VERSION="set-cookie.pl v1.0; #Jun 18 2002 Brain Force GmbH
-----#
#
# Simple set cookie demo.
#
#####
#- User configurable variables -----#
#set cookie name
$name = "cookie-name";
#set cookie value
$value = "demo-cookie";
#set expiry date for the cookie
$expDate = "09-Nov-2002 00:00:00 GMT";
#set this to your domain prepended with a .
$domain = ".qms.de";
#set path for the cookie
$path = "/";
#set to one if you want the cookie to be sent over a secure connection(ssl)
$secure = "0";
-----#
#main command to set cookie on a client side
&setCookie($name, $value, $expDate, $path, $domain);

# be sure to print a MIME type AFTER cookie headers and follow with a blank line
print "Content-type: text/html\n\n";

%cookies = %getCookies; # store cookies in %cookies
-----#

#- Set Cookie -----#
sub setCookie {
    # end a set-cookie header with the word secure and the cookie will only
    # be sent through secure connections
    local($name, $value, $expiration, $path, $domain, $secure) = @ ;

    print "Set-Cookie: ";
    print ($name, "=", $value, "; expires=", $expiration,
           "; path=", $path, "; domain=", $domain, "; ", $secure, "\n");
}
-----#

#- Retrieve Cookies From ENV -----#
sub getCookies {
    # cookies are separated by a semicolon and a space, this will split
    # them and return a hash of cookies
    local(@rawCookies) = split (/; /,$ENV{'HTTP_COOKIE'});
    local(%cookies);

    foreach(@rawCookies){
        ($key, $val) = split (/=,$ );
        $cookies{$key} = $val;
    }

    return %cookies;
}
-----#

```

3. MIB-tree

A file describing the MIBs (one4net.mib.desc) can be found on the CD.

II. Installation and Configuration Guide

A. Before Getting Started

You only need to connect via COM+ (Console) port with a terminal emulation application on PC to initially setup your LoadMaster machine(s).

Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM port (COM+ settings should 115200,8,N,1). After boot, a login prompt appears; login as 'bal' (password '1fourall'), and your Access Code will be displayed on the screen.

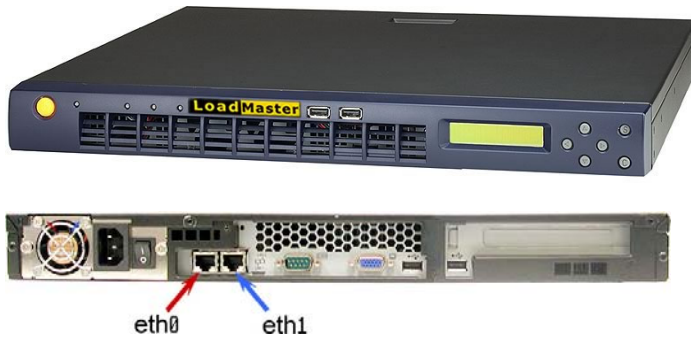


1. The LoadMaster Appliance

1.1. Delivery Content

- The delivery of each LoadMaster contains the following components:
- A/C power cable
- A CD containing the LoadMaster Software and the manuals in digital form.
- Rack mounts for standard 19" server racks (where applicable)

1.2. LoadMaster 1500 Hardware



LM-3620 and SM-1020 Hardware:

Ports:

2X10/100/1000Mbps LAN Ports
1XCOM+ Console Port
2XUSB Port

Dimensions:

16.7" (W) X 15.3" (D) X 1.7" (H)

Weight: 15.4lbs (7kg)

Input Voltage: 100 VAC-240 VAC
Full Range

Frequency: 47-63 Hz,

2. Connecting the Hardware

Connection of eth0

To the default gateway

Connect one end of a Category 5 Ethernet cable into the LAN port mapped to "eth0" and connect the other end to the hub/switch which interfaces with the default gateway.

Connection of eth1

Connect one end of a Category 5 Ethernet cable into the LAN port mapped to "eth1" and connect the other end to the hub/switch which interfaces with the server farm.

Note: Only the eth0 will point to the Network Side of the balancer. All other interfaces will default to the Farm Side of the balancer.

B. Initial Setup of your LoadMaster Single Unit (non-HA)

1. Login and License Key

Enter "bal" for the first login and "1fourall" for the password.

Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM port on the right, note the left COM port will not be used to connect to unit (COM+ settings should 115200,8,N,1).

Upon successful login a screen will appear with the following message:

"Thank you for purchasing LoadMaster. Please contact your KEMP representative to receive a License Key and unlock your LoadMaster".

Note down the Access Code and then contact you KEMP representative to get the License Key.

Note: When typing in the License Key, the keyboard has a US/ASCII mapping. The minus ("-") character on the numeric keypad will always work when "NumLock" is active.



During Quick Setup, you will be given the opportunity to change this setting to correctly reflect the actual keyboard layout being used.

Hint: You must have a service agreement or an evaluation window with KEMP Technologies to receive the Access Code of the LoadMaster.

License keys are linked to LoadMaster hardware and are not transferable.

Once a valid License Key has been input, Quick Setup will be started. For more information on Quick Setup, please consult the "Quick Setup" section.

C. Initial Setup of a LoadMaster High Availability (HA) Cluster

1. Login and License Key

Enter "bal" for the first login and "1fourall" for the password.

Using a null modem cable connect a PC using terminal emulation software from its COM+ port to the LoadMaster COM port on the right, note the left COM port will not be used to connect to unit (COM+ settings should 115200,8,N,1).

Upon successful login a screen will appear with the following message:

"Thank you for purchasing LoadMaster. Please contact your KEMP representative to receive a License Key and unlock your LoadMaster".

Note down the Access Code and then contact you KEMP representative to get the License Key.

Note: When typing in the License Key, the keyboard has a US/ASCII mapping. The minus ("-") character on the numeric keypad will always work when "NumLock" is active.

During Quick Setup, you will be given the opportunity to change this setting to correctly reflect the actual keyboard layout being used.

Hint: You must have a service agreement or an evaluation window with KEMP Technologies to receive the Access Code of the LoadMaster.

License keys are linked to LoadMaster hardware and are not transferable.

Once a valid License Key has been input, Quick Setup will be started. For more information on Quick Setup, please consult the "Quick Setup" section.

2. Configuring the second LoadMaster

Log in on the console of the second machine. Another Access Code will be displayed, obtain a second license key from your KEMP Representative as before and enter it on the console.

The LoadMaster will then request the local IP address that is to be used on Ethernet eth0 as well as the IP address of the first machine.

The machine will then transfer the configuration from the first machine. On completion the following message will be displayed:

"Most of the configuration parameters have been received from the partner LoadMaster. Only the local network interfaces must now be configured."



The local network interface addresses must now be input. The common IP address and network are displayed as a guide.

Once the local interface addresses have been input, the configuration should be activated. The HA cluster is now configured and ready for work.

If the parameters cannot be received from the partner, the full quick setup procedure must be followed using the same parameters (except local interface addresses) as on the first LoadMaster.

Upon rebooting the second machine, the configuration parameters will be overwritten by the values on the first machine.

Hint: Both real IP's as well as the shared IP addresses may be "pinged" to test the LoadMaster cluster.

D. Quick Setup

When you log into the LoadMaster for the first time and the license key has been validated the Quick Setup will start immediately.

Quick setup can also be accessed from the main configuration menu.

Quick setup allows a LoadMaster to be quickly configured; only the most important parameters needed by the LoadMaster are setup. Once the LoadMaster is configured and running, all the parameters can be changed using the configuration menu system.

Quick Setup welcomes you with the following message:

"This menu will allow you to quickly set up the balancer. The first step is to set up the network interfaces, then the hostname(s) of your LoadMaster(s) and finally the default gateway and DNS parameters."

The Quick Setup procedure allows the configuration of the following parameters:

- Ethernet IP address(s) – for eth0
- Ethernet IP address(s) – for eth1
- Hostname(s) – for local (and partner machine if running in a HA cluster)
- DNS parameters
- Domain parameters
- Default Gateway

After these parameters have been set, the configuration should be activated. The LoadMaster is then ready for work.

Note: If a parameter has been incorrectly set. Use the [CANCEL] button until the main menu appears. Quick Setup can then be performed again to correct the error.

Ethernet IP address(s) – eth0

The user is asked to input the IP address of the eth0 (NETWORK side) Ethernet interface. This should be input as a "dotted quad" followed by a network specifier.

I.e. 192.168.200.12/24

If no network specifier is given, the user will then be asked to specify the netmask, this may be input as either a network specifier (I.e. for the above example /24.) or as a "dotted quad" (I.e. If the IP address is 192.168.200.12 then the network mask should be 255.255.255.0).

A VLAN tag can now be entered for this interface. If no VLANs are to be used, this value should not be set. A VLAN tag can have a value between 1 and 4095.

Warning: A value of 1 is not recommended since it can cause networking problems with other manufactures equipment.



When configuring a HA cluster, the shared IP address will then be requested. This must be on the same network as the primary IP address of eth0 (as previously configured).

Ethernet IP address(s) – eth1

The user will now be asked to input the IP address of the eth1 (FARM side) Ethernet interface. When running in a Single-Armed configuration, this entry should be left empty.

The format of the input is the same as used for eth0. If an address is given, then this must be on a different network to the address(s) on eth0.

Hostname(s)

The hostname of the LoadMaster must now be set. A standard (or previously set) name is suggested. This name does not need to be changed unless the LoadMaster is to be part of a HA cluster and is to be installed on the same broadcast network as other LoadMaster clusters.

When configuring a LoadMaster HA cluster, the name of the partner machine is requested, a standard name is also suggested here. This name also does not need to be changed unless the configuration requires it.

DNS configuration

The DNS resolver is now configured. Up to three DNS servers may be specified (Addresses must be in “dotted quad” syntax).

A list of search domains can now be given. Up to 6 domains can be specified.

E. Main Menu

Many features of the LoadMaster can be configured using the menu system. The menu system can be used by logging onto the console as “bal”, or by remotely logging into the system using the SSH protocol.

Important: Remote access is only permitted if the SSH service is enabled and the password for “bal” has been changed from its default value. If the password has not been changed from its default value, the user “bal” will only be allowed to login from a directly connected console.

Note: If the password for “bal” has been forgotten, a user can login on the console as **pwreset**. The password is **1pwreset**, this will reset the password for “bal” to **1fourall** until the LoadMaster is rebooted. If unit is rebooted the password will be reset to its old (unknown) value. It is thus strongly advised that the password should be changed using the configuration menu before the next reboot.

1. Configuration Menu basics

The configuration menu system is made up of a number of hierarchical menus split into functional groups. Navigation around the menus can be performed by using the *Up* and *Down* cursor keys, or by using the “+” and “-” keys. On menus with numeric entries, the number can also be given.

Example: To change the keyboard mapping, the user can type 3<CR> - which selects the “Local Administration” menu, followed by 3<CR> for “set keyboard map”.

Using “q”<CR> or “ESCAPE” or using the [CANCEL] button will return the user to the previous menu.

Hint: To access the [OK], and [CANCEL] buttons, use the TAB key to toggle between the menu and the buttons.

Using the [CANCEL] button from the main menu, all changes made to the configuration will be ignored.



Using the [OK] button from the main menu performs the menu point, which is currently highlighted.

Important: When the LoadMaster is configured in a HA cluster, and the user is logged onto the standby machine, only the configuration of the local IP interfaces, changing the local password and performing a backup/restore should be performed, all other configuration parameters should only be changed on the active machine. From the main menu, the following options are available.

1.1. Quick Setup

This allows the user to quickly configure the basic parameters of the LoadMaster, these include the Ethernet IP addresses and local gateways and name servers.

See the section on “quick setup” in the initial configuration section.

2. Service Management (CLI)

This menu point starts a CLI (command line interface), which lets the user to administer the Virtual Services that are available on the LoadMaster. See section III for information on the syntax of the commands.

To leave the CLI, the user can type “exit”, or use the ESCAPE or CTRL-D keys.

In this version of the LoadMaster, the syntax of CLI commands has been changed. The original syntax may be selected using the menu option “Use MML format CLI” under Utilities -> Diagnostics.

3. Local Administration

This menu performs administration tasks for the current LoadMaster balancer. The following options are available:

3.1. Set Password

Using this option, the user may change the local password for the user “bal”. The password should be changed for security reasons. Remote access over SSH is not allowed until the password has been changed.

Important: The password is not saved when performing a backup and is not replaced when performing a restore.

If the LoadMaster is running in a HA (high availability) mode cluster. Each LoadMaster can have a separate password. The password information is not transferred between the members of a cluster.

3.2. Set Date/Time

This option allows the local date, time and time zone to be set.

A list of time zones is given; the current time zone is always at the start of the list. The user may select a different time zone it required.

The date should be entered in the following format:
02-12-03 (Year-Month-Day)

Followed by the time in the following format:
10:57:15 (Hours:Minutes:Seconds)

Note: When first delivered the LoadMaster is set to use UTC.

3.3. Set Keyboard Map

This option allows the keyboard mappings to be changed to support different languages. A list of different keyboard mappings is supplied; the current mapping is always at the start of the list.



Note: The default keyboard mapping is US/ASCII.

Changes to keyboard mappings do not have any affect during an SSL session. Only after reconnection will the keyboard mappings be activated.

After a keyboard mapping has been selected, the user will be asked to check that the keyboard mapping is correct. If the keyboard mapping is not correct the [CANCEL] button should be pressed and a different mapping selected.

3.4. Backup/Restore

This option allows the configuration of the LoadMaster to be saved to either to a remote machine.

When using remote backup, the backup server machine must run an FTP daemon or an SSH daemon.

When performing a restore (from a remote machine), the user may select what information should be restored:

Only the Virtual Service configuration

Only the information about the virtual services will be restored.

Only the LoadMaster Base Configuration

Only the LoadMaster configuration not including the virtual service configuration.

Both the Virtual Service and Base Configuration information

All the configuration information on the LoadMaster.

Important: Restoring the Virtual Service Configuration on the standby LoadMaster of a HA cluster is not permitted since the Virtual Service configuration is always taken from the Active LoadMaster, and this would overwrite any restored configuration.

3.5. Remote Access Control

This option allows the user to enable or disable remote access to the LoadMaster.

Enable/Disable Remote SSH access

This option allows enables or disables access to the LoadMaster via the SSH protocol. If this option is disabled, the menus can only be accessed via the local console. If no password has been specified for "bal", it is not possible to log in via SSH.

Enable/Disable Remote Web access

This option enables or disables access to the Web user interface.

Change Web Address

The LoadMaster is delivered with the Web user interface configured to be only accessible via the "network" side address. With this option, the Web user interface can be configured to be accessible from only a "farm" side address.

4. Basic Setup

This menu allows the user to perform each of the steps in the "quick setup" separately.

4.1. Network configuration

The configuration of the various IP addresses of the Ethernet interfaces can be configured.



When using the LoadMaster in a one-armed configuration, the second interface does not have to be configured. When asked to configure the second interface (eth1) just press the [OK] button with no IP address supplied.

If the LoadMaster is supplied with extra optional Ethernet interfaces, these interfaces can only be configured using this menu. In this case, the on-board interfaces are no longer eth0 and eth1 but the highest numbered Ethernet interfaces. I.e. the optional interfaces will be designated as eth0 and eth1. For more information on this topic please contact customer support.

4.2. Hostname Configuration

The hostname of the LoadMaster can be changed. When the system is configured as a HA cluster, the hostname of the partner LoadMaster can also be changed.

Hint: It is not required to change the name of the LoadMaster unless there are multiple HA clusters on the same broadcast network (Ethernet segment).

4.3. DNS configuration

This option allows the configuration of the LoadMaster name resolution facility. If no DNS parameters are specified, the administration of the LoadMaster must be performed using "dotted quad" addressing only.

This option allows the configuration of up to three DNS server addresses. These must be in "dotted quad" format.

Up to 6 search domains may also be specified.

4.4 Routing Configuration

This option permits the configuration of default and static routes.

The LoadMaster requires a default gateway through which it can communicate with the Internet. See the "Application Guide" section for more information on this subject.

Other routes can also be specified using this menu. These routes are static and the gateways must be on the same network as the LoadMaster.

5. Extended Configuration

This menu allows the user to configure several features, which do not directly affect the main function of the LoadMaster but makes the balancer easier to use.

5.1. Interface Control

This option allows the configuration of the protocol used at the physical level on the Ethernet. Normally, the LoadMaster will auto detect (auto-negotiation) which Ethernet protocol it should use. Sometimes this process does not always work. With this option, the user may force the LoadMaster to use a specific protocol (either 100Mb Full or Half Duplex).

Note: If the LoadMaster is connected to a 10Mbit switch, then auto detect MUST be used.

5.2. Enable/Disable S-NAT

This toggle option will either enable or disable the S-NAT functionality of the LoadMaster. When S-NAT is enabled, the real servers can access the Internet using the LoadMaster as a gateway. The LoadMaster will use "masquerading" so that connection requests from the real servers seem to originate on the LoadMaster. This means that the real servers can be on a private network and still have access to the Internet.



When S-NAT is disabled, the LoadMaster will not perform “masquerading” and so the real servers cannot access the Internet through the LoadMaster.

In Single-Armed configurations, S-NAT does not provide any extra functionality.

5.3. Syslogd Configuration

With this option, log messages may be sent to different hosts using the syslogd protocol.

A different host may be specified for each of five different levels:

NOTICE	This host will receive all messages from the LoadMaster.
WARN	This host will receive all messages except NOTICE level messages.
ERROR	This host will receive all messages except WARN and NOTICE level messages. I.e. It will receive ERROR, CRITICAL and EMERGENCY messages.
CRITICAL	This host will receive only CRITICAL and EMERGENCY messages.
EMERGENCY	This host will receive only EMERGENCY messages.

5.4. SNMP metrics

With this menu, the SNMP configuration can be modified. For more information on SNMP please see the Application Guide.

Enable/Disable SNMP metrics

This toggle option, enables or disables SNMP metrics. I.e. This option allows the LoadMaster to respond to SNMP requests.

Note: By default SNMP is disabled.

Configure SNMP Clients

With this option, the user can specify from which SNMP management hosts the LoadMaster will respond to.

Important: If no client has been specified, the LoadMaster will respond to SNMP management requests from **any** host.

Configure SNMP Community String

This option allows the SNMP community string to be changed. The default value is “public”.

Configure SNMP Contact

This option allows the SNMP Contact string to be changed. For example, this could be e-mail address of the administrator of the LoadMaster.

Configure SNMP Location

This option allows the SNMP location string to be changed.

5.5. SNMP traps

When an important event happens to a LoadMaster a Virtual Service or to a real server, a trap is generated. These are sent to the SNMP trap sinks.

Enable/Disable SNMP Traps

This toggle option enables and disables the sending of SNMP traps.



Note: SNMP traps are disabled by default.

Configure SNMP Trap Sink1

This option allows the user to specify a list of hosts to which a SNMPv1 trap will be sent when a trap is generated.

Configure SNMP Trap Sink2

This option allows the user to specify a list of hosts to which a SNMPv2 trap will be sent when a trap is generated.

5.6. Enable/Disable L7 persistency state failover

Note: This feature is only available on a HA cluster configuration

When an L7 persistency option has been enabled, the active LoadMaster will automatically send connection information to the standby machine so that if the active machine fails, the standby machine can take over the processing of requests as if nothing had happened. The connection information is sent using a Multicast protocol. The parameters for this may be changed under "Multicast Configuration".

This toggle option will either enable or disable the transfer of L7 connection information. If this feature takes too much bandwidth or is not required, then it may be safely disabled.

5.7. Enable/Disable L4 connection state failover

Note: This feature is only available on a HA cluster configuration

When a Virtual Service is not using persistency or only IP source address persistency, the active LoadMaster will automatically send connection information to the standby machine so that if the active machine fails, the standby machine can take over the processing of requests as if nothing had happened. The connection information is sent using a Multicast protocol. Only the Ethernet interface parameter under "Multicast Configuration" has any affect on this option.

This toggle option will either enable or disable the transfer of L4 connection information. If this feature takes too much bandwidth or is not required, then it may be safely disabled.

5.8. Multicast Configuration

Note: This option is only available on a HA cluster configuration when the L7 persistency state failover or L4 connection failover features are enabled.

With this option, the multicast address and Ethernet interface, which is to be used for the transfer of connection information, can be changed. When using the LoadMaster in single-armed mode, the Ethernet interface cannot be changed.

5.9. HA timeout

Note: This option is only available on a HA cluster configuration.

With this option, the time it takes a HA cluster to detect a failure can be adjusted. A value between and 1 and 5 can be set. The default value is 1. A lower value will detect failures sooner, while a higher value gives better protection against a DOS attack.

6. Packet Filter & Access Control Lists

6.1. Access control Lists

The LoadMaster supports a "blacklist" Access Control List system. Any host or network entered into the Access Control List will be blocked from accessing any service provided by the LoadMaster.



The LoadMaster also has a packet filter. When enabled the packet filter blocks all IP packets which are not directed at a configured port.

The Access Control list is only enabled when the packet filter is enabled. By default the Access Control List is disabled. This means that all source IP addresses are accepted by the LoadMaster.

Enable Access Control Lists

Using this toggle option the Packet Filter/Access Control List can be activated / deactivated.

Show ACL

This option lists the content of the current Access Control List.

Add address to ACL

This option allows a user to add a host or network IP address to the Access Control List. Only "dotted-quad" IP addresses are allowed. A network is specified by using a network specifier.

I.e. Specifying 192.168.200.0/24 will block all hosts on the 192.168.200 network.

Delete address from ACL

This option allows an IP address or network to be deleted from the Access Control List.

Reject/Drop blocked packets

When a connection request is received from a host, which is blocked using the ACL, the request is normally ignored (dropped). The LoadMaster may however be configured to send back an ICMP reject packet. For security reasons it is usually best to drop any blocked requests.

7. Utilities

7.1. Software Upgrade

Using this option, patches for the operating software of the LoadMaster may be installed or removed.

Install Update

With this option, a patch can be downloaded onto the LoadMaster from a remote server. The server must be running a SSH daemon.

Once the patch has been downloaded, the patch is unpacked and verified. If the patch is valid, then the name of the patch will be displayed and the user will be asked to confirm if the patch should be installed. A copy of the current operating software is saved before the patch is installed, this may be recovered at a later date using the "rollback update" option.

Rollback Update

If a patch needs to be removed, this option allows the previous version of the operating software to be recovered. Only one previous version is available. When the software has been recovered, it is not possible to recover any earlier versions.

7.2. Transfer mode

This option allows the user to specify which transfer method should be used to transfer data between the LoadMaster and a remote server. The selected method is used to store a backup on a remote server or to download software patches. The default method is "ftp".



Use ftp protocol

Using this option, the Internet standard “ftp” protocol is used. Most servers support this protocol.

Use scp protocol

The “scp” - secure copy – transfer method may be selected. This is more secure than “ftp” but is normally only supported on UNIX servers. If this mode is selected, the transfer of SSL certificates can only be performed via the menu system and not via the Web interface.

Use http protocol

Using this transfer method, backups to a remote server cannot be performed. Software patches can however be downloaded from any Web server where the patch has been made available.

7.3. Network Time Protocol Host

The time on the LoadMaster can be synchronized to an NTP server. The time will be synchronized at boot time and every then on an hourly basis. Using this option, the address of the NTP server can be specified.

7.4. SSL certificate administration

This option permits the administration of currently installed SSL certificates. A list of virtual services, which have SSL acceleration enabled, is given. Selecting a virtual service allows the certificates for the service to be managed. Selecting the local option allows the certificate used for the Web interface to be regenerated.

Get a certificate file

This option allows the user to download a certificate file for the virtual service.
Note: the SCP protocol may be used to transfer certificate files.

Get a key file

This option allows the user to download a private key file for the virtual service. If a private key is included in the certificate file, no additional private key file is required.

Delete the key and certificate files

Allows the user to delete a certificate and key file for a specific virtual service.

7.5. Update License

This option permits the input of a new license key. I.e. when updating from an evaluation to a full license.

7.6. Diagnostics

This menu allows the user to perform diagnostic functions on the LoadMaster.

Ping Remote Host

A remote host may be “pinged”.

Enable Diagnostic login

Important: The option “Enable diagnostic login” should only ever be enabled when requested to by LoadMaster support staff.

If this option is enabled in normal operation, this may result in unauthorized access to the LoadMaster. The diagnostic login will be disabled upon reboot of the LoadMaster or it can be disabled from this menu.



Show Partner IP Address

If the LoadMaster is being used in a HA configuration and the real addresses of either partner is changed, it can cause both LoadMasters to no longer communicate with each other. This option allows the changing of the partners IP address so that communication can be restored.

8. Reboot

This option will reboot the LoadMaster. All modifications to the configuration will be saved before the reboot.

Note: When running on the active machine of a HA cluster, the configuration on the standby machine will also be updated before the standby machine becomes the active machine (Since the active machine is being rebooted).

9. Exit LoadMaster Config

This option allows the user to leave the configuration menu system.

If any parameters have been changed, the user will be asked if they want to activate the changes. If this is confirmed, then the changes are activated. If the user does not want to activate the changes, the user will be asked if they want to save the changes for a later activation. If this is NOT confirmed, all changes will be lost.

F. The LoadMaster Questionnaire

1. Single LoadMaster Balancer Solution

Machine 1

Network side: eth0
IP Address

Netmask

Farm side: eth1
IP Address

Netmask

Hostname

Name Servers
(Space separated list)

Search Domains
(Space separated list)

Default Gateway
(IP Address)

2. Highly Available dual LoadMaster Balancer Solution

Machine 1

Machine 2

Network side: eth0
IP Address



Netmask

Shared IP address

Farm side: eth1
IP Address

Netmask

Shared IP address

Hostname

Name Servers
(Space separated list)

Search Domains
(Space separated list)

Default Gateway
(IP Address)

III. Command Line Interface

Reference Guide

The command interface syntax is loosely based on the industry standard syntax as used by other Load Balancer manufacturers.

The command interface has a line based, hierarchical command set. Changes made to the configuration are only performed when returning to the top level.

Hint: A port can either be specified as a numeric value or as a symbolic name. The following names are recognized:

DNS	53
FTP	21
HTTP	80
IMAP4	143
LDAP	389
POP2	109
POP3	110
SMTP	25
SNMP	161
SSL	443
TELNET	23
TFTP	69

1. Top level commands

At the top level the following commands may be specified.

1.1. Adaptive

This command switches the input to the adaptive parameters command set.



1.2. Delete <name|VIP>

This command will delete the specified VIP.

1.3. Disable_rs <IPspec>

This command will disable the specified Real Server. I.e. No more traffic will be directed to the Real Server. This command will disable the Real Server on all virtual services where this Real Server is configured.

1.4. Enable_rs <IPspec>

This command will re-enable the specified Real Server. The Real Server will be re-enabled for all virtual services.

1.5. Health check

This command switches the input to the health check parameter command set.

1.6. Rules

This command switches the input to the rule configuration command set. Rules are only available if the L7 option has been activated.

1.7. Show <name|VIP>

This command will display all information about the given Virtual Service. If no Virtual Service is specified, information about all Virtual Services will be displayed.

1.8. Vip <name|VIP>

This command switches the input to the virtual service command set. A <VIP> is the IP address of the Virtual Service. A <name> is the name of the Virtual Service.

If no Virtual Service with the specified IP address (or IP name respectively), then a new Virtual Service will be created. No changes will occur to the configuration until the user returns to the top level command level.

1.9. Help

Prints a summary of commands at the current level.

1.10. End

Terminate the CLI session.

1.11. Exit

Since the input level is at the top level, this command has no affect.

2. Adaptive scheduling command level

The following commands are available at the adaptive command level. No changes to the configuration will occur until the command level returns to the top level. I.e. when the user types "exit".

2.1. Interval <Integer>

With this command, the interval of sampling the server loads will be set to <Integer> seconds.

2.2. Min <Integer>



The minimum load (as a percentage) where adaptive balancing takes effect can be set. If the mean load of the server falls below this threshold, the virtual service will be considered "idle" and the weights will return gradually to their "static" values.

2.3. Port <PortSpec>

The specified port will be used to access the Real Servers where adaptive checking is enabled.

2.4. Show

Displays the current adaptive checking parameters.

2.5. Url <String>

<String> specifies a URL, which will be fetched by the adaptive checking system. The contents of this URL should specify the load on the current Real Server, with 0 representing no load and 100 representing a fully loaded server.

See section " IF.5 Agent Based Adaptive Balancing" for more details.

2.6. Weight <Integer>

This specifies the minimal value of the weight (as a percentage of the static weight). The adaptive scheduling method will not adjust a server weight below this value.

2.7. Help

Prints out a list of the available commands at the adaptive command level.

2.8. End

Terminates the CLI session. No changes performed after entering this level will be saved.

2.9. Exit

Returns the input to the top command level. Any changes will be written to the configuration file, and the system will be updated accordingly.

3. Health check command level

The following commands can be performed at the health check command level.

3.1. Interval <Integer>

Specifies how often the health of a Real Server should be checked.

3.2. Retry <Integer>

Specifies how often the health check of a Real Server should fail before the LoadMaster decides that the Real Server is no longer responding.

3.3. Show

Displays the current health check parameters.

3.4. Timeout <Integer>

Specifies how long the LoadMaster should wait for a response from a Real Server. The LoadMaster will mark a Real Server as down after Timeout * Retry seconds if no response has been received.

3.5. Help



Lists the commands that are available at the health check command level.

3.6. End

Terminate the CLI session. Any changes since entering the health check command level will be ignored.

3.7. Exit

Leave the health check command level, any changes to the health check parameters will be saved and the system will be configured accordingly.

4. Rules command level

The following commands can be performed at the rules command level.

4.1. Add <Rule-name>

This command creates a new rule <Rule-name>. It also switches into the Rule Edit command level. Upon return to the Rules command level. Further rules may be added. A rule must be added before a Real Server can use it.

4.2. Modify <Rule-name>

This command switches into the Rule Edit command level, so that the rule <Rule-name> can be edited.

4.3. Delete <Rule-name>

This deletes the specified rule. The rule will be deleted from all Real Servers to which it has been assigned.

4.4. Show [<Rule-name>]

Displays a list of all the rules (if no <Rule-name> parameter) is specified or the specified rule.

4.5. Help

Lists the commands that are available at the rules command level.

4.6. End

Terminate the CLI session. Any changes since entering the health check command level will be ignored.

4.7. Exit

Leave the rules command level, any changes to the rules will be saved and the system will be configured accordingly.

5. Rule Edit command level

The following commands can be performed at the rule edit command level.

5.1. value <string>

This option allows the match string value of the rule to be set. Spaces are significant. By default a string is treated as a regular expression. If <prefix> or <postfix> is set, then the string is treated as a literal string, which is then matched at the start or end of the received URL respectively.

5.2. [no] negation

This command inverts (reverts to normal if [no] is specified) the sense of a rule. I.e. If negation is set, the rule will be true if the received URL does NOT match the value of the rule.



5.3. [no] prefix

This specifies that the value of the rule should be matched at the start of the received URL.

5.4. [no] postfix

This specifies that the value of the rule should be matched at the end of the received URL.

5.5. [no] regex+host

This specifies that the value of the rule should be matched against the concatenated hostname and received URL string.

5.6. [no] prefix+host

This specifies that the value of the rule should be matched at the start of the concatenated hostname and received URL string.

5.7. [no] postfix+host

This specifies that the value of the rule should be matched at the end of the concatenated hostname and received URL string.

Note: If no prefix or postfix option is enabled, the default rule matching will be a regular expression. Specifying **no** to any of the above options reverts the matching back to regular expression matching without any hostname concatenations.

5.8. Show

Displays the value of the current rule.

5.9. Help

Lists the commands that are available at the rule edit command level.

5.10. End

Terminate the CLI session. Any changes since entering the rules command level will be ignored.

5.11. Exit

Leave the rule edit command level and return to the rules command level. Modifications will not be saved until after the rules command level is "exited".

6. Virtual Service (VIP) command level

The following commands are available at the Virtual Service command level. No changes will be made to the system until the user performs an "exit" from this level. If the VIP has errors, the user will be asked if the VIP should be discarded. If the VIP is discarded, the input will return to the top level. If the VIP is not discarded, the input will remain at the Virtual Service command level, the user may then correct the error.

6.1. [no] Adaptive <String>

Specifies whether the Virtual Service should support adaptive health checking. The only current method is "http_rs". To disable adaptive health checking for a Virtual Service, the command <no adaptive> should be used.

6.2. Add <IPspec>



This command adds the Real Server as specified by the <IPspec> to the Virtual Service. It also switches the input into the Real Server command level. Upon return from the Real Server command level, further Real Servers can be added to the Virtual Service.

6.3. Address <IPspec>

Specifies the IP address of the Virtual Service.

6.4. [no] Cookie <String>

Allows the specification of a cookie when using Cookie based persistency methods. This command can only be used if the L7 option of the LoadMaster has been enabled. When using the <passive-cookie> and <passive-cookie-src> persistency modes, the cookie string is mandatory. To delete a cookie string, use the command <no cookie>.

6.5. Delete <IPspec>

Deletes a Real Server as specified by <IPspec> from the Virtual Service. A Virtual Service must have at least one Real Server.

6.6. Disable

Disable the Virtual Service. This means that the Virtual Service will accept no new requests.

6.7. Enable

Re-enable a Virtual Service. The Virtual Service will again accept new requests.

6.8. Follow <Port Spec>

This command only works if the L7 option of the LoadMaster has been enabled. This specifies

6.9. Mask <Ipmask>

When using L4 (source IP based persistency), An IP mask may be specified which is used to determine if two IP addresses should be treated as coming from the same source. By default the mask has a value of 255.255.255.255, which means that all IP addresses are different.

6.10. [no] Name <Name>

Specifies the "name" of the Virtual Service. To delete the name use the command <no name>.

6.11. Healthcheck <String>

This specifies which health-check method should be used for a given Virtual Service. If the Virtual Service has a well-known port, a health check method will be automatically set. The following health check methods may be specified.

http	Http checking is enabled
https	Https (SSL) checking is enabled
smtp	The (simple mail transfer protocol) is used.
nntp	The (network news transfer protocol) is used.
ftp	The (file transfer protocol) is used.
telnet	The (telnet protocol) is used.
pop3	The (postoffice – mail client protocol) is used.
imap	The (imap – mail client protocol) is used.
tcp	A basic TCP connection is checked.
dns	A DNS request is sent to the Real Servers port. This checking method is only valid when using a UDP protocol.



udp A dummy zero length UDP packet is sent to the port.
icmp An ICMP ping is sent to the Real Server.

6.12. [no] Persist <Persist type>

This command specifies which type of connection persistence should be used for a Virtual Service. In no persistency should be specified for the Virtual Service, the command <no persist> should be specified. The following persistency types can be specified. If the L7 option has not been enabled, only the <src> persistency is allowed.

ssl	The Session ID in an SSL connection is used to maintain client to real server persistency.
cookie	Server generated cookies will be used.
active-cookie	LoadMaster generated cookies will be used.
url	A request for a specific URL will always go to the same real server.
host	A request to the same virtual host will go to the same real server.
src	Enables IP based persistency.
cookie- src	Server generated cookies will be used. If the client does not return a cookie, the clients' IP address will be used.
active- cook-src	A LoadMaster generated cookie will be used. If the client does not return the cookie, the clients' IP address will be used.
cookie- hash	All connections with the same set of cookies will always be sent to the same real server. If no cookies are sent, normal scheduling will occur.

6.13. Port <Port spec>

Specifies the IP port to be used for the Virtual Service. If no health check mechanism has been specified and the port is a well-known port, the relevant health check mechanism will be selected.

6.14. Precedence <rule-name> <number>

The precedence of the rule <rule-name> is set to <number>. A value of 1 moves the rule to the start of the rule list. I.e. this rule is checked first. A higher value moves the rule to the respective position in the rule. If a <default> rule is specified for a Real Server, its precedence will always be lower than any user defined rules. I.e. a <default> rule will always be checked after every other rule.

6.15. Protocol <tcp/udp>

Protocol to be used for the Virtual Service. This may be <tcp> or <udp>. By default the protocol will be set to <tcp>.

6.16. Ptimeout <Integer>

Specifies how long the LoadMaster should remember the persistency information associated with a connection. This value is specified in seconds.

6.17. Schedule <schedule method>

This allows the scheduling method between the Real Servers to be specified. The following scheduling methods may be specified:

rr	round robin (default).
wrr	weighted round robin.
lc	least connection.
llc	weighted least connection.

6.18. Server <IPspec>

This command enters the Real Server command level for the specified Real Server. The Real Server must already be assigned to the Virtual Service.



6.19. Show

Displays all the parameters of the current Virtual Service.

6.20. Help

Prints out a list of commands at the Virtual Service command level.

6.21. End

Terminate the CLI session. No changes made in the Virtual Service command level (or lower) will be saved.

6.22. Exit

Return the input to the top level. Any changes to the Virtual Service will be saved. If an error is detected in the Virtual Service, the system reports the error and asks if the Virtual Service should be discarded. If the Virtual Service is not discarded, the input remains at the virtual service level, where any corrections may be made.

7. Real Server command level

At this command level, a specific Real Server may be configured. The following commands are available at this level.

7.1. Addrule <Rule-name>

This command adds the rule <Rule-name> to a Real Server. If this is the first assignment of <Rule-name> to a Real Server on the current Virtual Service, the rule will be placed on the precedence list as the lowest user defined rule i.e. checked after all other rules. Use the Virtual Service command **Precedence** to change the precedence order.

7.2. Delrule <Rule-name>

This command removes the association of rule <Rule-name> from the Real Server. If there are no more instances of the rule associated with the Virtual Service, the rule will be deleted from the Virtual Service precedence list.

7.3. Disable

Disables the current Real Server. The Real Server will only be disabled in the current Virtual Service. If the Real Server is accessed via a different Virtual Service, then this Virtual Service will not be affected.

7.4. Enable

Re-enable the current Real Server on this Virtual Service. If the Real Server has been disabled on multiple Virtual Services, these Virtual Services will not be affected.

7.5. Forward <forwarding method>

This specifies the forwarding method, which should be used to access the Real Service. This can be either <nat> or <route>. By default the forwarding method is <nat>, <route> should only be selected when using "direct service return".

7.6. Port <portspec>

Specifies which port on the Real Server should be used. If no port is specified, then the port from the Virtual Service will be used.



7.7. Show

Display the parameters for the current Real Server.

7.8. Weight <integer>

Specifies the weighting for the Real Server. This can be used when using the various scheduling methods that utilize the weighting of a Real Server.

7.9. Help

Lists the commands at this level.

7.10. End

Terminate the CLI session. No changes made in the VIP and Real Server command levels will be saved.

7.11. Exit

Return to the Virtual Service command level. No changes will be saved until the editing of the current Virtual Service has been completed.



IV. Web User Interface (WUI) Configuration Guide

A. Glossary and Abbreviations

Access Code: An Access Code will be generated during the initial setup of the Load Master. You must contact your KEMP Technologies representative for your 60-day evaluation or your full purchased license key.

Balancer: A network device or logic that distributes inbound connections with a common source address across a farm of server machines.

Farm Side: The Load Master network interface to which the server farm is connected.

Flat-based: The VIPs and the real servers are on the same subnet.

HA: Highly Available or High Availability (used interchangeably)

ICMP: Internet Control Message Protocol

MIB: Management Information Base, a database of object definitions. The definition specifies whether an SNMP manager can monitor the object.

NAT: Network Address Translation

NAT-based: The VIPs and the real servers are on different subnets.

Network Side: The Load Master network interface over which requests to the server farm are made.

One-armed: Only one Ethernet interface is used for in and outbound traffic. Farm side and Network side are both connected to it.

RS: Real Server: Physical server machines which make up a server farm.

Service: A Service is an application that is connected to the network.

Shared IP: The shared (floating) IP address is always the assigned IP address of the active Load Master in a HA solution.

SCP: Secure copy command of SSH

SNMP: Simple Network Management Protocol, a network protocol used to manage TCP/IP networks. This protocol provides functions that enable you to access the data object whose definitions are located in the MIB.

S-NAT: Network Address Translation for a source IP address.

SSH: Secure Shell Protocol

Two-armed: Two Ethernet interfaces are used for in and outbound traffic, one connected to the network side and one to the farm side.

UTC: Universal Time Coordinated

VIP: Virtual IP Address: The IP address of a service defined on the Load Master.

VS: Virtual Service: An entry on the Load Master over which a service being hosted in the server farm can be reached.

WUI: Web User Interface used to perform Load Master administration via a web browser.



B. Fast Track

The following sections will take you through the steps required to create virtual services of increasing complexity.

1. How To Login

Start your preferred Internet browser and enter the URL of the Balancer that you want to manage.

Then you are asked to authenticate. The default username is 'bal' with the pre-defined password '1fourall'.

Note: A password for user 'bal' must be set in Initial Set-up on the console of the Balancer. That password will be the one that will be used to connect to WUI.



Enter Network Password

Please type your user name and password.

Site: 192.168.0.62

Realm: user

User Name: bal

Password:

Save this password in your password list

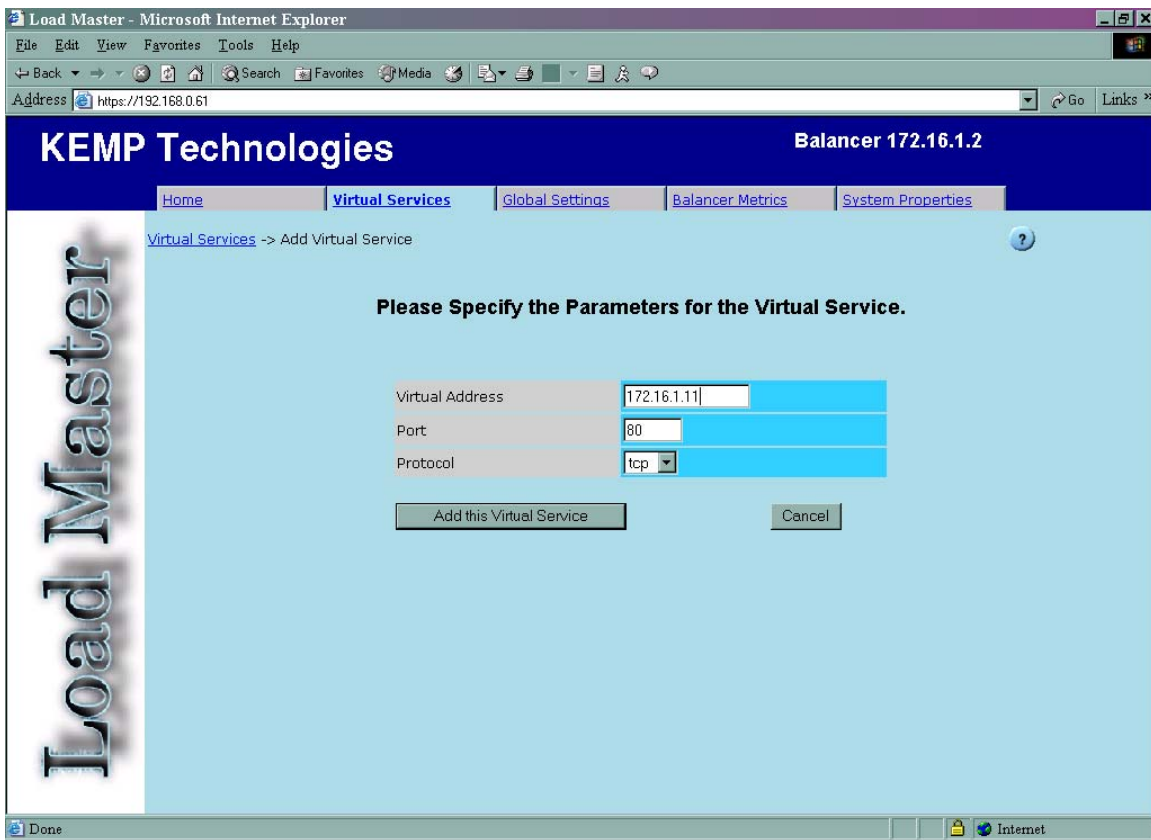
OK Cancel

2. Create a Simple Virtual Service

This section will take you through the steps required to create a simple virtual service that has two real servers.

Firstly, click on the "Virtual Services" tab to bring up the virtual service page. Any virtual services that are on the balancer are listed here and their properties are summarized. To begin the process of creating a new virtual service, click the "Add Virtual Service" button. This brings up the virtual service parameters page and it is here that you enter the virtual IP (VIP) address of your virtual service, its port and the protocol.





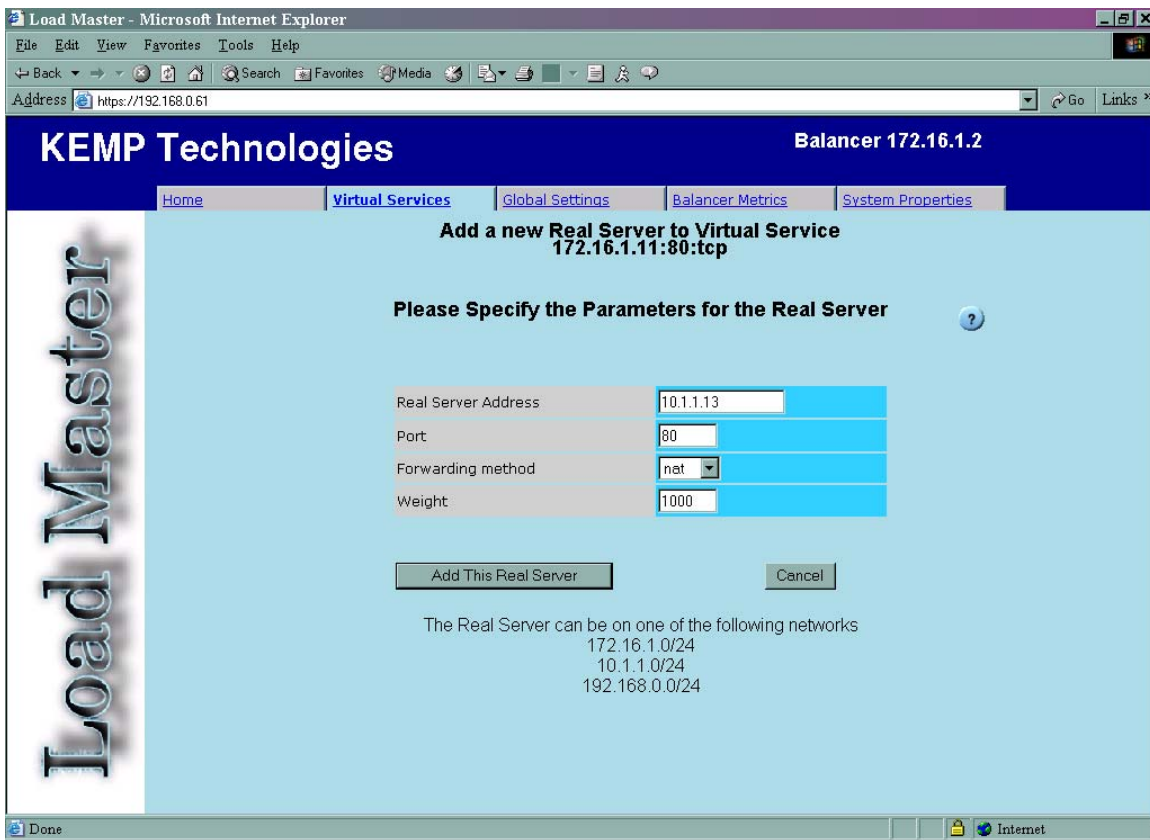
For example, if you gave your customer “www.a-domain.com” the IP address 172.16.1.11 then enter this as the VIP address. The port number is usually 80 for http services. The protocol may be TCP or UDP, but in the vast majority of cases TCP will be the one used.

Once you are satisfied with the choice of VIP, port and protocol click “Add This Virtual Service” to bring up the virtual service properties. In this example, we are not concerned with most of these values and will create a virtual service with no persistence, no content switching and Round Robin as the scheduling method, which are the default settings so nothing needs to be altered.



The final action to be performed is adding real servers. To get to the real server parameters page, click the "Add New..." button in the real server table. Here we specify the IP address of the real server we wish to add, the port and forwarding method it is to use and its relative weight.





In our example, we will assume that the real servers are on a private 10.1.1.x network, and we will then enter real server 10.1.1.13. At this stage, we do not need to worry about the port, forwarding method and weight. Click "Add This Real Server" to finish.

The virtual service properties page should now display the recently added real server in the real server table. To add another real server, repeat the process but with a different real server IP address.

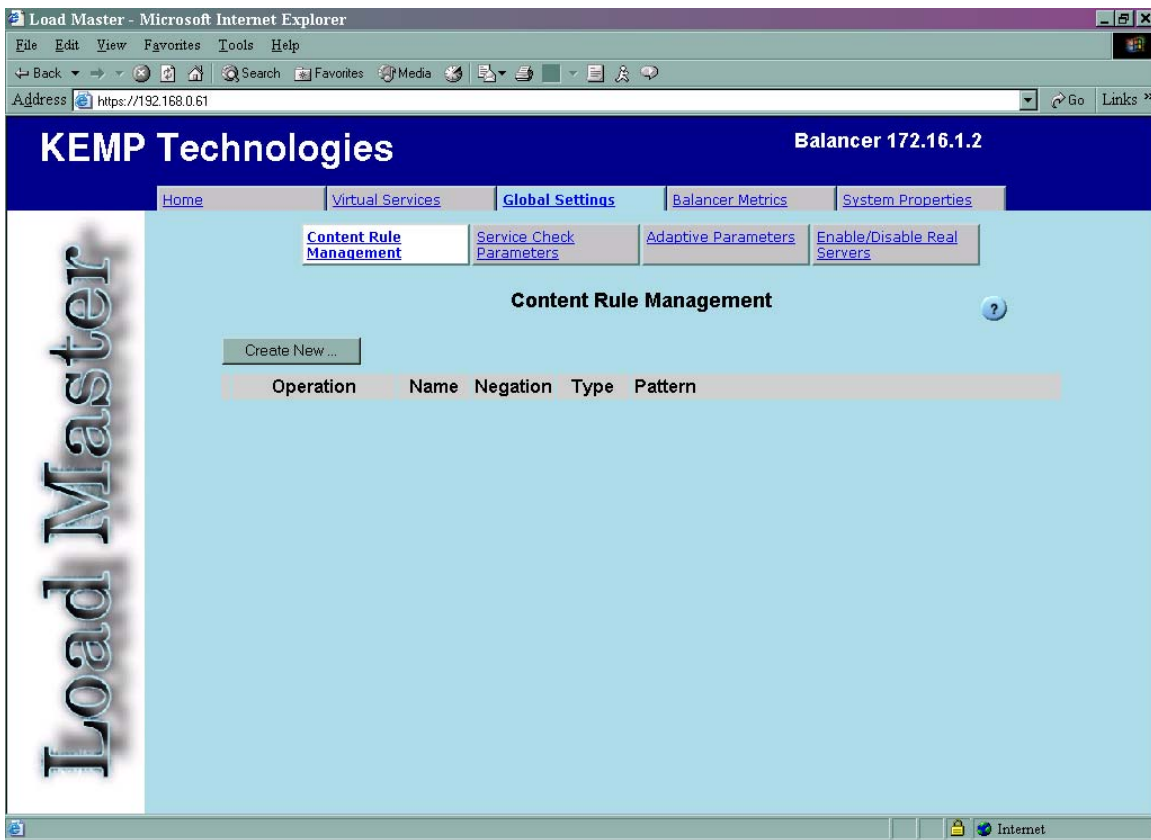
All changes are made in real-time, so we have now created the virtual service. To return to the virtual service page, either click "Virtual Services" link at the top-left of the frame, or click the "Virtual Services" tab. The virtual service table should now list the service we have just created.

3. Create a Virtual Service with Content Rules

This section will take you through the steps required to set up a virtual service that makes use of content switching. Content Switching means that the Balancer can distribute requests to a server depending on the content of the request.

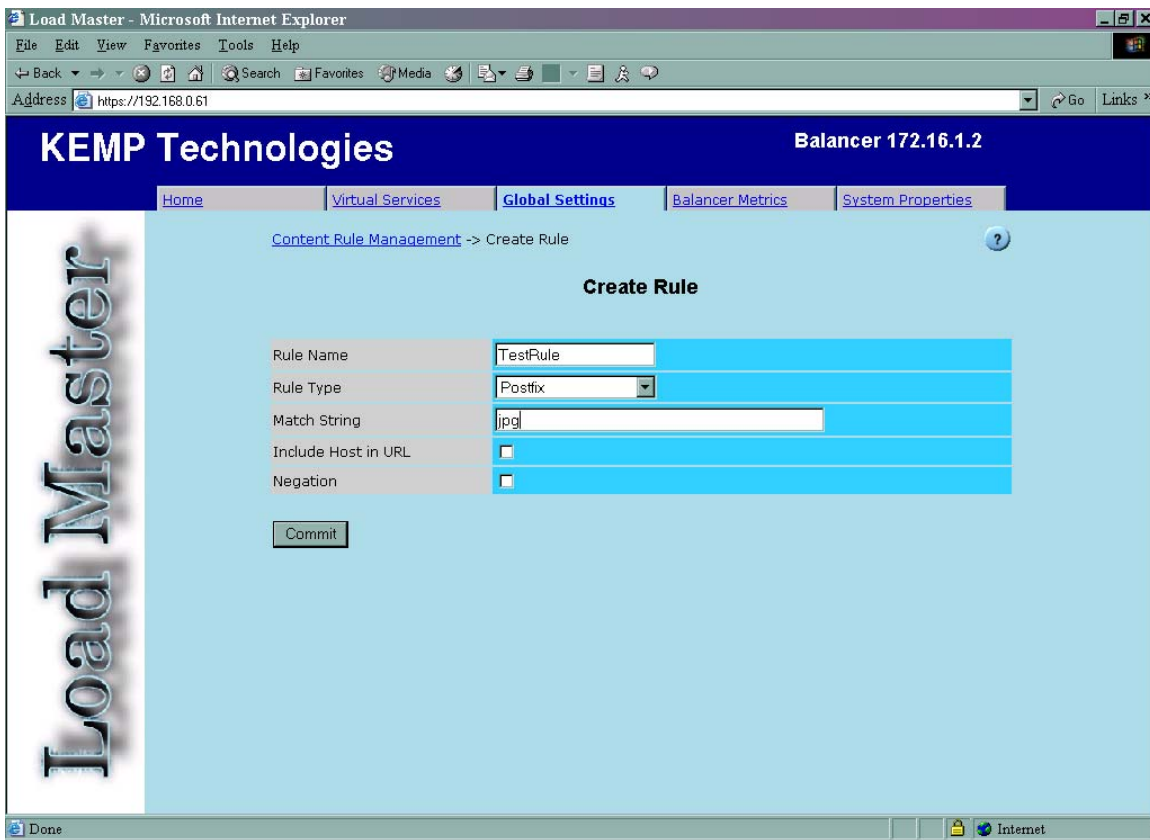
To begin with, we need to create at least one content rule. It usually makes sense to create more rules. The content rule management section of the WUI is found under the "Global Settings" tab. Clicking on this tab brings up the "Content Rule Management" section as default. This section consists of a summary list of content rules, if they exist.





To create a new content rule, click “Create New...” to open the rule creation page. There are five parameters that can be set for a rule, but only “Rule Name”, “Rule Type” and “Match String” are mandatory and for the purposes of this example we will not use the other two, “Include Host in URL” and “Negation”. In the “Rule Name” field, enter the name by which the rule should be known. In this case we will call our rule “**TestRule**”. Next, select the type of rule: Prefix, Postfix or Regular Expression. A description of these rule types may be found in the next chapter, and in this example we will choose “Postfix”. Finally, enter the text string that the balancer will attempt to match, in this example we will enter “jpg”, so we will perform content switching on requests for JPEG graphics. Click “Commit” to finish and return to the “Content Rule Management” page, where the rule just created should be listed.



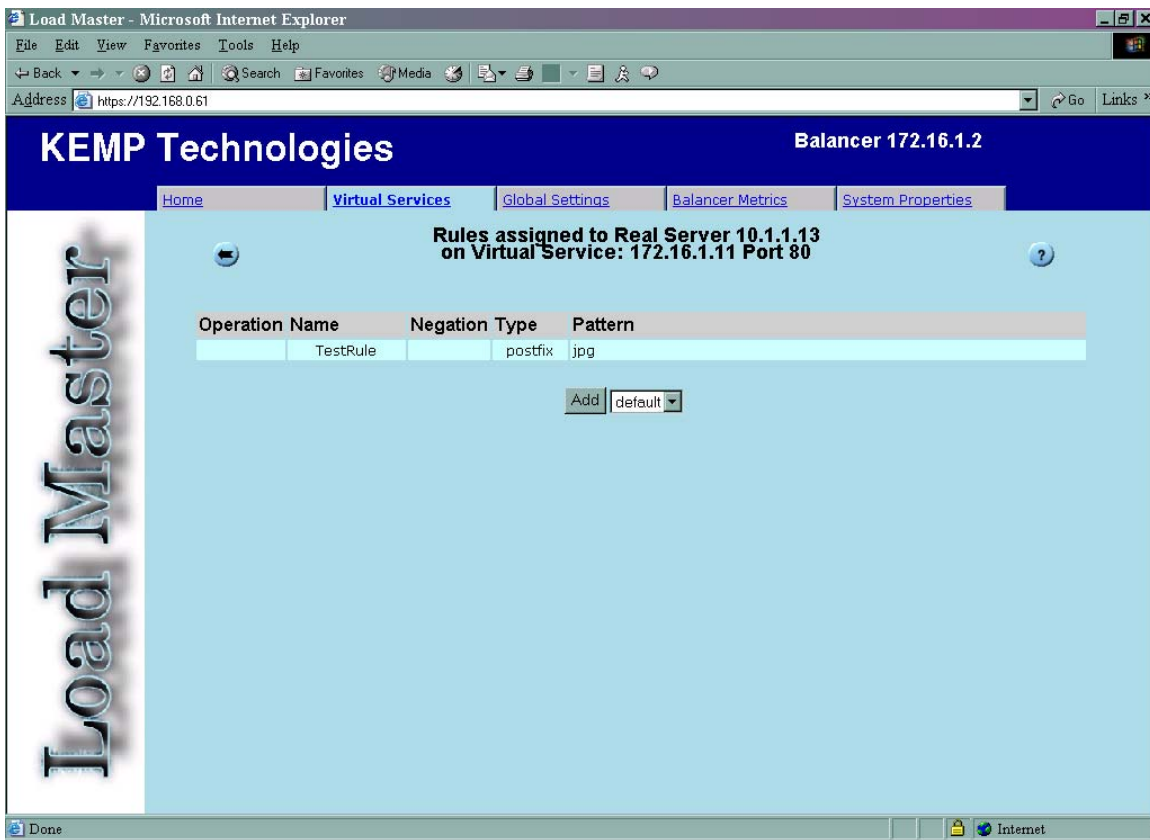


The next step is to enable content switching for a virtual service. If you wish to create a new rule with content switching, follow the steps outlined in the previous section for creating a virtual service, specifying its VIP, port and protocol. If you wish to add content switching to an existing virtual service, click that service's "Modify" button. On the properties page, you will notice the text "Content switching: disabled" in the "Content Switching and Persistence Options" section.

Add a real server, in the way described in the previous section, and you should notice that there is now a button entitled "Enable" in the "Content Switching and Persistence Options" section. Click this button, and an extra column, "Rules", will appear in the "Real Servers" table containing a button that displays the number of rules assigned to that real server. In this case, the button will display "None". Click the button to add rules to the real server. For example, if real server 10.1.1.13 contained all JPEG files we would wish to add the "TestRule" to this real server.

The rule assignment page shows a summary list of rules that are assigned to the real server in question and a pull-down list of rules that have already been defined, and have not been assigned to this real server yet. There is also a rule that is always present, entitled "default" and this rule basically governs all requests that do not match any other selected rules. Select the rule defined earlier (in our case "TestRule" and click "Add". It should now appear in the summary list. Click the arrow button in the top left of the page to return to the virtual service properties page. Repeat the rule assignment process for all real servers - those that do not require a specific rule must be assigned the "default" rule otherwise they will effectively be useless, and depicted as "Down".





Content switching has now been enabled for this virtual service. Click on the "Virtual Services" link at the top left of the screen to return to the "Virtual Services" page.

4. Create an SSL accelerated Virtual Service

This section will explain how to create a Virtual Service with SSL Acceleration activated.

SSL Acceleration transfers the processing of SSL from the real servers to the balancer, meaning that only one certificate is required per virtual service.

Note: When SSL Acceleration is enabled, communication from the balancer to the real servers is unencrypted.

Firstly, create a new service (see first section) that has the port 443 (HTTPS). Make certain that the persistence is not set to SSL and a Real Server has been assigned to this service. Simply check the SSL Acceleration checkbox to enable SSL Acceleration.

If there is no certificate for that virtual service, you will be prompted to install a certificate. To download a certificate, enter the remote host where the certificate is located and your username and password for this host. Then enter the filename of the certificate and the private key, and click "Get File" to install them.





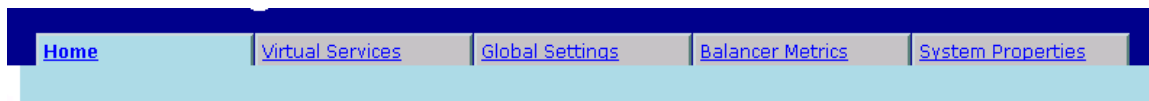
There are a couple of points to note about the consequences of enabling SSL Acceleration:

1. It sets the port value of this virtual service's real servers to 80.
2. It sets the service check method to HTTP and not HTTPS as would normally be the case with SSL services.

C. Full Menu Tree

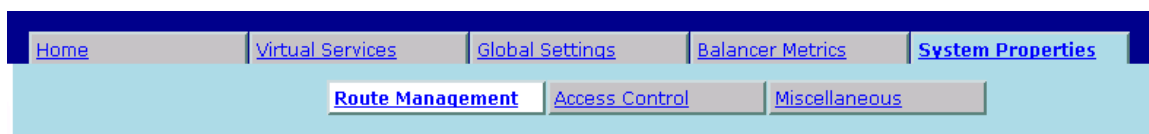
This section is Quick Reference that shall help you find your way through the menu structure of the Load Master WUI.

The Balancer menu consists of navigation tabs on the upper side of the screen:

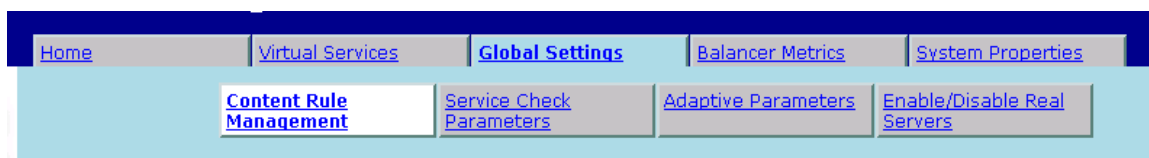


Two of these tabs each have a submenu.

These two tabs are 'System Properties'



and 'Global Settings':



1. Home

An introduction page.

2. Virtual Services

A list of virtual services on the balancer, summarizing the properties of each and giving the options to modify or delete services, or create a new service.

2.1. Add Virtual Service

Here the Virtual IP (VIP) address, port and protocol are defined. The VIP address and port must be typed manually into the text fields and the protocol is selected from the pull-down list.

2.2. Virtual Service Properties

Here the properties of the virtual service are set, and the real servers are added (or removed).

Activate or Deactivate Service

This checkbox gives you the option to activate or deactivate a virtual service. The default is checked - active.

Service Check for the Virtual Service

This provides a list of checks for well-known services, as well as lower level checks for TCP/UDP or ICMP. With the service checks, the real servers are checked for the availability the selected service. With TCP/UDP the check is simply a connect attempt.

The following health-check methods may be specified.

http	Http checking is enabled
https	Https (SSL) checking is enabled
smtp	The (simple mail transfer protocol) is used.
nntp	The (network news transfer protocol) is used.
ftp	The (file transfer protocol) is used.
telnet	The (telnet protocol) is used.
pop3	The (postoffice – mail client protocol) is used.
imap	The (imap – mail client protocol) is used.
tcp	A basic TCP connection is checked.
dns	A DNS request is sent to the Real Servers port. This checking method is only valid when using a UDP protocol.
udp	A dummy zero length UDP packet is sent to the port.
icmp	An ICMP ping is sent to the Real Server.

Service nickname

This text field allows you to assign a nickname to the Virtual Service being created, or change an existing one.

Content Switching and Persistence Options

This section allows you to select whether persistence is enabled for this service, to set the type of persistence and the persistence timeout value, and to enable or disable content switching.

If persistence is enabled it means that a client connection to a particular real server via the balancer is persistent, in other words the same client will subsequently connect to the same real server. The timeout value determines for how long this particular connection is remembered.



The pull-down list gives you the option to select the type of persistence. These are:

IP-based Persistence (SRC)

The source IP address (of the requesting client) is used as the key for persistency in this case. The netmask determines how the Balancer sees a 'client' in the context of persistence. For example:

When the netmask is set to 255.255.255.255 (the default) then every individual IP address qualifies as a valid persistent context. So, a client with the IP address 200.190.125.67 connects, and its connection to a particular real server is remembered. The client then ends the session and disconnects. A short time later, the client begins another session, but this time its IP address is given as 200.190.125.44 - it will not necessarily be directed to the same real server as before.

However, using the example above, if the netmask is set to 255.255.255.0, then all clients connecting with an IP address of 200.190.125.X will be grouped together and directed to the same real server, until the timeout has expired.

URL Persistence

Requests to the same URL go to the same server.

Cookie Persistence

The Balancer checks the value of a specially set cookie in the HTTP header. Connections with the same cookie will go to the same real server.

(Standard) Cookie

The real server must be configured to set the special cookie.

(Standard) Cookie-Src

If cookie persistence fails, it reverts to source-based persistence.

Active Cookie

The Balancer automatically sets the special cookie.

Active Cookie-Src

If active cookie persistence fails, it reverts to source-based persistence.

Host Persistence

A request to the same host always goes to the same server.

SSL Persistence

The SSL session ID is used to keep a session connected to the same server.

Note: SSL acceleration cannot be used with this method.

Port Following

Port following enables a switch from an HTTP connection to an HTTPS (SSL) connection to be persistent on the same real server. Port following can only be switched on if the current service is an HTTPS service, and if there exists a HTTP service with the same IP address as this HTTPS service.



Scheduling method

This section allows you to select the method by which the balancer will select a real server, for this particular service. The scheduling methods are as follows:

Round Robin

Round Robin causes the balancer to assign real servers to a session in order, i.e. the first session connects to real server 1, the second to real server 2 etc. There is no bias in the way the real servers are assigned.

Weighted Round Robin

This method uses the weight property of the real servers to determine which real servers get preference. The higher the weight a real server has, the higher the proportion of connections it will receive.

Least Connection

With this method, the current real server with the fewest open connections is assigned to the session.

Weighted Least Connection

As with Least Connection, but with a bias relative to the weight.

Adaptive

Adaptive scheduling means that the load on the real servers is periodically monitored and that packets are distributed such that load will be approximately equal for all machines. More detail can be found in the section covering Global Settings.

SSL Acceleration

This checkbox appears when the criteria for SSL Acceleration have been met, and serves to activate SSL Acceleration. For more information about SSL Acceleration, please refer to the Fast Track C chapter. If there is no certificate for the virtual service, you will be prompted to install a certificate. To download a certificate, enter the remote host where the certificate is located and your username and password for this host. Then enter the filename of the certificate and the private key, and click "Get File" to install them.

2.3. Real Server Assignment

This section lists the real servers that are assigned to the virtual service. The properties of the real servers are summarized and there is also the opportunity to add or delete a real server, or modify the properties of a real server. When Content Switching is enabled, there is also the opportunity to add rules to, or remove rules from, the real server (see Add Rule).

2.4. Add / Modify Real Server

Here, the properties of the real server are set. These are:

The real server IP address (this is not editable when modifying a real server).

The forwarding port of the real server. This field is editable, so the port may be altered if necessary.

The forwarding method. This is NAT - Network Address Translation - or Route (Direct) forwarding – if available dependent on the other modes selected for the service.

The real server's weight. This is weight of the real server, as used by the Weighted Round Robin, Weighted Least Connection and Adaptive scheduling method. The default initial value for the weight is 1000, the maximum is 65535, the minimum is 1. It is a good benchmark to give a real server a weight relative to its processor speed, i.e. if server1 seems to bring four times the power of server2, assign a weight of 4000 to server1 and weight of 1000 to server2.



2.5. Add Rule

This contains a summary list of rules assigned to the real server in question. Add a rule by selecting it from the pull-down list and clicking "Add", remove a rule by using the delete button. See chapter IE.1.

2.6. Rule Precedence

This contains a summary list of rules assigned to the Virtual Service in question. A rule may be promoted in the order of precedence by clicking its corresponding "Promote" button.

3. Global Settings

3.1. Content Rule Management

To define a new rule, click on "Create New". You must give the rule a name.

Hint: Rule names must be alphanumeric and start with a character.

Note, however, that rule names are unique and case sensitive, which means that giving a rule an existing name will overwrite the rule of that name, and also that two different rules can exist in the form "Rule1" or "rule1".

Next, decide the type of rule this will be: Prefix, Postfix or Regular Expression. These match to the URL as follows:

```
/absolute/pathname/of/the/url/foo.html  
|-> Prefix                                   Postfix ->|  
|->       Regular Expression               ->|
```

With the "Include host in URL" checkbox checked, the host name is also included in the URL match string:

```
www.a-host.com/absolute/pathname/of/the/url/foo.html  
|-> Prefix                                   Postfix ->|  
|->                                       Regular Expression               ->|
```

The protocol definition (e.g. http://) is ignored in all cases. Finally, enter the string that is to be matched. Prefix and Postfix use standard strings and match exactly to what is entered. Regular Expression uses the following syntax:

- ? - Match any single character
- * - Match zero or more characters
- \$ - End of line
- ^ - Start of line
- [- Start of set, of which only one character will be matched. Must be terminated by]
 - ^ At start of a set matches a character not in the set.
- \ - Escapes the next character

3.2. Service Health Check Parameters

For further information on Health Checking please refer to chapter Kin the Installation Guide.

3.2.1. Check Interval

With this field you can specify the number of seconds that will pass between consecutive checks. The recommended value is 7 seconds.



3.3. Connect & Response timeouts

The HTTP request has two steps: contact the server, and then retrieve the file. A timeout can be specified for each step, i.e. how long to wait for a connection, how long to wait for a response. A good value for both is 3 seconds.

3.4. Re-try Count

This specifies the number of retry attempts the check will make before it determines that the server is not functioning. A value of 1 or less disables retries.

4. Adaptive Parameters

4.1. Adaptive Interval

This is the interval, in seconds, at which the balancer checks the load on the servers. A low value means the balancer is very sensitive to load, but this comes at a cost of extra load on the balancer itself. 7 seconds is a good starting value. This value must not be less than the HTTP checking interval (see below).

4.2. Adaptive URL

The Adaptive method retrieves load information from the servers via an HTTP inquiry. This URL specifies the file where the load information of the servers is stored. The standard location is "/load". It is the servers' job to provide the current load data in this file in ASCII format. In doing so, the following must be considered:

An ASCII file containing a value in the range of 0 to 100 in the first line (where 0=idle and 100=overloaded). The file is set to "/load" by default.

The file must be accessible via HTTP

The URL must be the same for all servers that are to be supported by the adaptive method

Note: This feature is not only of interest for HTTP based Virtual Services, but for all Services. HTTP is merely used as the transport method for extracting the application specific load information from the Real Server.

4.3. Port

The port number of the HTTP daemon on the servers. The default value is 80.

4.4. Min Control Variable Value

This value specifies a threshold below which the balancer will switch to static weight-based scheduling, i.e. normal Weighted Round Robin. The value is a percentage of the maximum load (0-50 max.). The default is 5.

4.5. Min Weight Adjustment Value

This value specifies the minimum weight that can be assigned to this server, as a percentage of the initial static weight it was given. For example, if the server was given 50 to start with and this adjustment value is 10, then the weight will never be set to below 10. A value of 5 is recommended.

4.6. Real Server Availability

This section displays the current online status of real servers, and enables a real server to be disconnected or shut down cleanly. Each real server has a corresponding button, and pressing this button will take an online server offline, and vice-versa.



5. Balancer Metrics

The balancer metrics sections provide performance data relating to the Balancer, and are updated every 15 seconds to provide near real-time information. This can be very useful when tuning the load balancing of your server farm for optimal performance, general trouble-shooting and error detection. Furthermore, this feature is invaluable for making performance comparisons across Virtual Services and Real Servers.

5.1. Global Metrics

CPU

This graph displays the following CPU utilization information for a given Balancer:

Use	the percentage of the CPU, which is spent in processing in user mode
System	the percentage of the CPU spent processing in system mode
I/O Waiting	the percentage of the CPU spent waiting for I/O to complete
Idle	the percentage of CPU, which is idle

Note: User + System + I/O Waiting + Idle = 100%

Memory

This bar graph shows the amount of memory in use and the amount of memory free on the balancer.

Network Activity

These bar graphs show the current network throughput on each interface.

5.1.2. Real Server Metrics

These graphs display the connections, bytes or packets (depending on choice: the buttons in the top right of the page toggle which value is to be displayed) handled by each real server. The value is a sum over all virtual services that this real server is a part of, and is represented as a percentage of the overall value for the whole balancer.

5.1.3. Virtual Service Metrics

These graphs display the total number of connections (or bytes or packets) for each virtual service, and displays how these are distributed across the virtual service's real servers by means of the percentage of the total for the virtual service that each real server handles.

6. System Properties

6.1. Route Management

This option permits the configuration of default and static routes.

The Load Master requires a **default gateway** through which it can communicate with the Internet. Further routes can be added. These routes are static and the gateways must be on the same network as the Load Master.

6.2. Access Control

6.2.1. Packet Filter Enabled

Using this toggle option the Packet filter can be activated/deactivated. If the filter is not activated, the Load Master acts as a simple IP-forwarder. When the filter is activated, only the Virtual Service addresses can be addressed.



6.2.2. Reject/Drop blocked packets

When a connection request is received from a host, which is blocked using the ACL, the request is normally ignored (dropped). The Load Master may however be configured to send back an ICMP reject packet. For security reasons it is usually best to drop any blocked requests.

6.2.3. Access control Lists

The Load Master supports a “blacklist” Access Control List system. Any host or network entered into the Access Control List will be blocked from accessing any service provided by the Load Master. The Access Control List is only enabled when the Packet Filter is enabled.

6.2.4. Add Address

This option allows a user to add a host or network IP address to the Access Control List. Only “dotted-quad” IP addresses are allowed. Using a network specifier specifies a network.

I.e. Specifying 192.168.200.0/24 will block all hosts on the 192.168.200 network.

6.3. Miscellaneous

6.3.1. SNAT Control

This toggle option will either enable or disable the S-NAT functionality of the Load Master. When S-NAT is enabled, the real servers can access the Internet using the Load Master as a gateway. The Load Master will use “masquerading” so that connection requests from the real servers seem to originate on the Load Master. This means that the real servers can be on a private network and still have access to the Internet.

When S-NAT is disabled, the Load Master will not perform “masquerading” and so the real servers cannot access the Internet through the Load Master.

In Single-Armed configurations, S-NAT does not provide any extra functionality.

6.3.2. Set Transfer Protocol

This option allows the user to specify which transfer method should be used to transfer data between the Load Master and a remote server. The selected method is used to store a backup on a remote server, to download software patches or to manage certificates. The default method is “ftp”.

Use ftp protocol

Using this option, the Internet standard “ftp” protocol is used. Most servers support this protocol.

Use scp protocol

The “scp” - secure copy – transfer method may be selected. This is more secure than “ftp” but is normally only supported on UNIX servers.

Use http protocol

Using this transfer method, backups to a remote server cannot be performed. Software patches can however be downloaded from any Web server where the patch has been made available.

Note: The scp protocol cannot be used for transferring certificate files over WUI.

6.3.3. Set HA Timeout

This option is only available on HA systems.

With this option, the time it takes a HA cluster to detect a failure can be adjusted. A multiplier between and 1 and 5 can be set. The default value is 1. A lower value will detect failures sooner, while a higher value gives better protection against a DOS attack.

