Performance Benchmarking

March 2006

Contents

The purpose of this whitepaper is to present an overview of the performance benchmark testing performed as part of the overall "Quality Assurance" testing program for the MailMarshal 6.0 SMTP development program.

The results of this testing can be used to calculate the likely requirements for a new MailMarshal installation in terms of MailMarshal server specifications, database size and bandwidth requirements.

The results detailed here were used to produce the MailMarshal 6.0 SMTP Sizing whitepaper which is also available. Please note that any recommendations included within this whitepaper are presented as a recommendation only. There are a number of assumptions made in this whitepaper when calculating server loadings which must be understood before applying thecalculations to a new installation.

Careful attention should still be paid to the deployed solution to ensure it is operating at an acceptable level.

Authors: Peter Hodges and Bradley Anstis

MARSHAL
Secure. Protect. Comply.

Abstract

The goal of this whitepaper was to provide a measure of the performance of MailMarshal SMTP 6.0 under several common customer configurations. The evaluation was performed using typical modern server grade hardware that would be readily available to a customer.

At the end of this whitepaper there is also guidance given on configurations, scaling and the calculations used to determine these values. This information can be used to determine the recommended configuration for new MailMarshal 6.0 SMTP deployment.

Test Environment

The test environment was designed to simulate a real world installation where MailMarshal sits between a companies internal mail servers and the internet (fig 1).

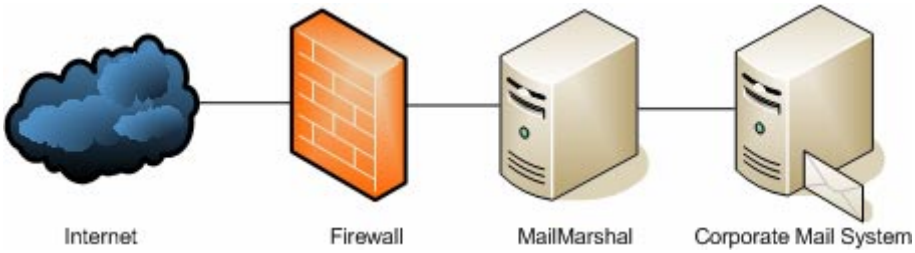


Fig 1 – Typical placement of MailMarshal in the Email flow

In order to produce sufficient email volume both the corporate mail system and the Internet were modeled using three Microsoft Windows 2003 servers each.

The corporate firewall typically just provides a direct routed connection to the internet over a fixed bandwidth pipe; this was modeled using a Microsoft Windows 2003 server running as a router with bandwidth shaping software installed.

All servers were interconnected using full duplex 100Mbs Ethernet to a switch. Figure 2 describes the complete test environment

MARSHAL
Secure. Protect. Comply.

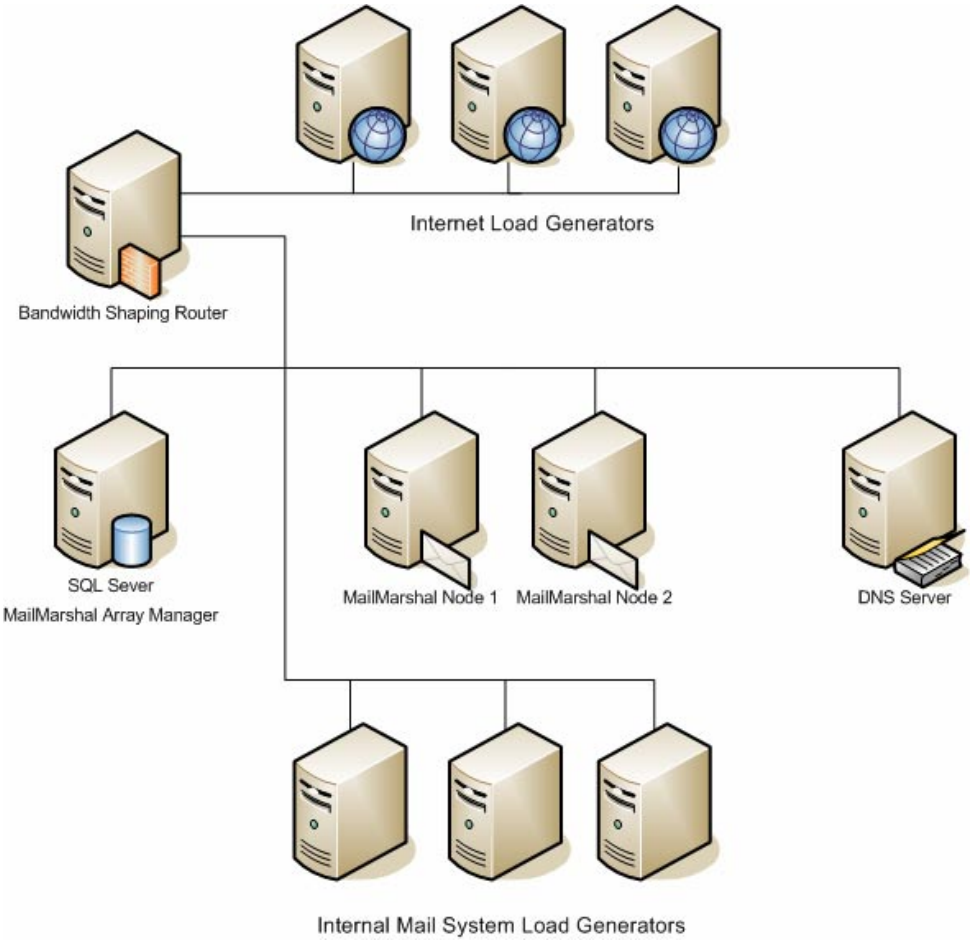


Figure 2 – Performance Benchmarking Test Environment

In order to generate the test load from the Internet and Local servers, an in-house Email generation tool,"Bench-Tool" was used. Bench-Tool operates by reading rfc822 formatted emails out of a source directory,rewriting the "To:" and "From:" fields then sending the message to the server under test (see appendix for further information).

MailMarshal build 6.0.0.156 was used for this testing. All servers had Microsoft Windows 2003 server installed. The default MailMarshal SMTP 6.0 rules were used which included rules for Spam classificationinto suspect and explicit folders and anti-virus scanning (a total of 36 rules, 18 inbound, 10 outbound and 8 bidirectional).

Sophos was used for the anti-virus engine.

The MailMarshal Nodes were Dual 2.8G Xeon Compaq DL360 servers with 2 x 146G disks and 2G of memory (1U form factor).

The MailMarshal Array Manager / SQL server was a Dual 2.8G Xeon Compaq DL380 server with 2 x 146G and 4x 72G disks and 4G of memory (2U form factor).

MARSHAL
Secure. Protect. Comply.

Test Data

The test data was obtained from a 1000 user company. The data was selected from a random time period of mail received on a Wednesday. The resultant data consisted of 3969 incoming and 2496 outgoing messages.

Approximately 50% of the incoming email is spam, and 2 messages contained viruses.

	Incoming	Outgoing
Number of Messages	3969	1496
Average Size	30100 bytes	34800 bytes
Messages < 10K	86%	84%
Messages > 1M	0.8%	0.4%

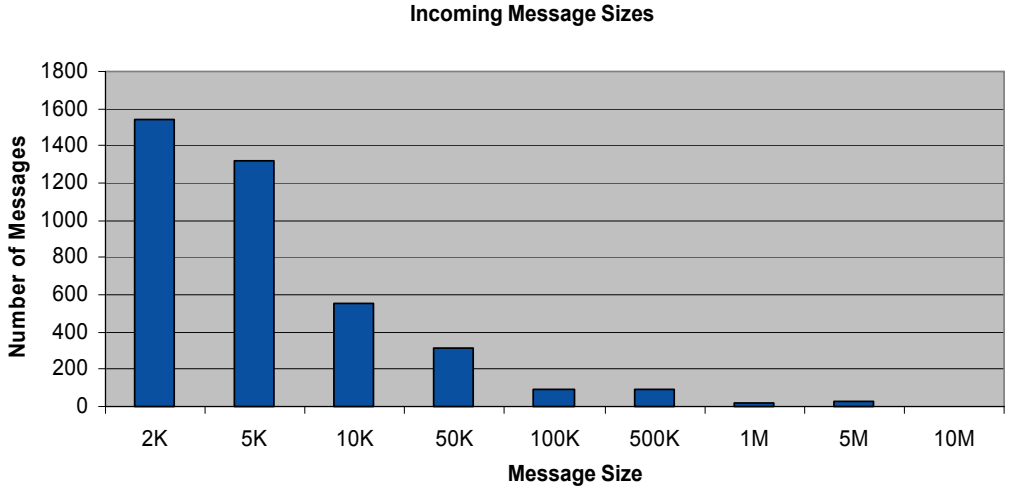


Figure 1 - Incoming message size breakdown

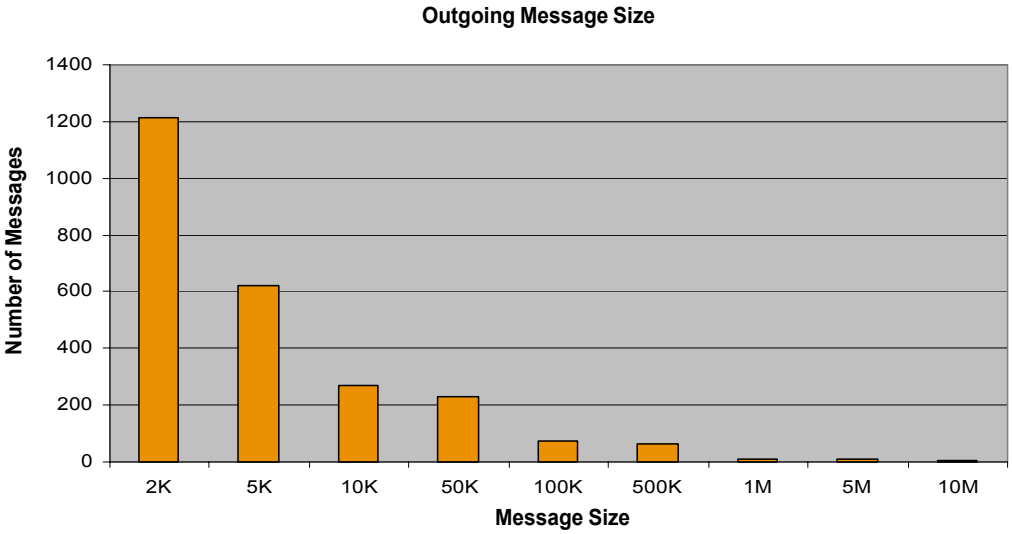


Figure 2 - Outgoing message breakdown

The Bench-Tool running on each of the load generating machines was configured to send to one or more random recipients. The distribution of the number of recipients was modeled after that observed on our own Auckland based Marshal mail server (94% to one user tailing off to a maximum of 32 users). Each recipient name was composed of a name with random number between 1 to 20,000 and a domain with random number between 1 and 3.

This yielded 60,000 unique local user names and 60,000 unique Internet names.

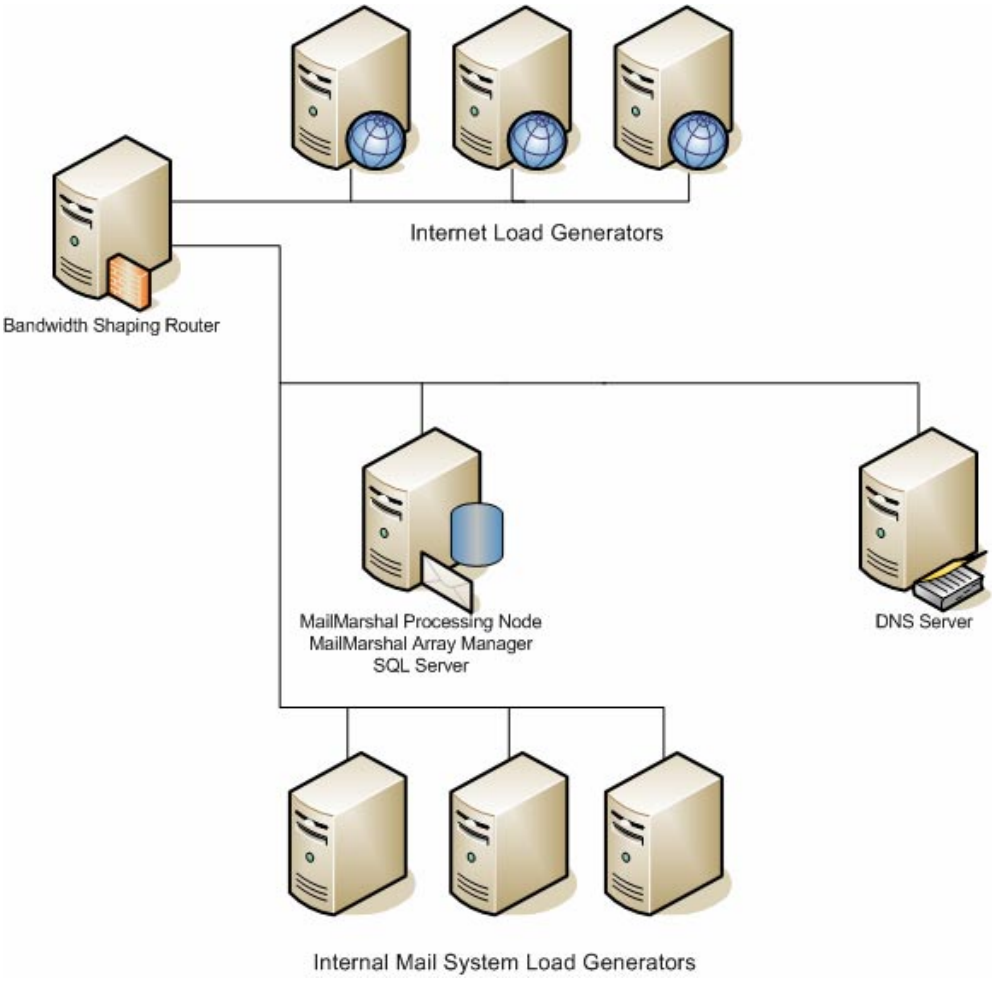
The domain names were chosen so that emails from the Internet servers would route (via a Local Domain entry) to an internal server, and email from internal servers would route via DNS MX lookup to an Internet machine.

MARSHAL
Secure. Protect. Comply.

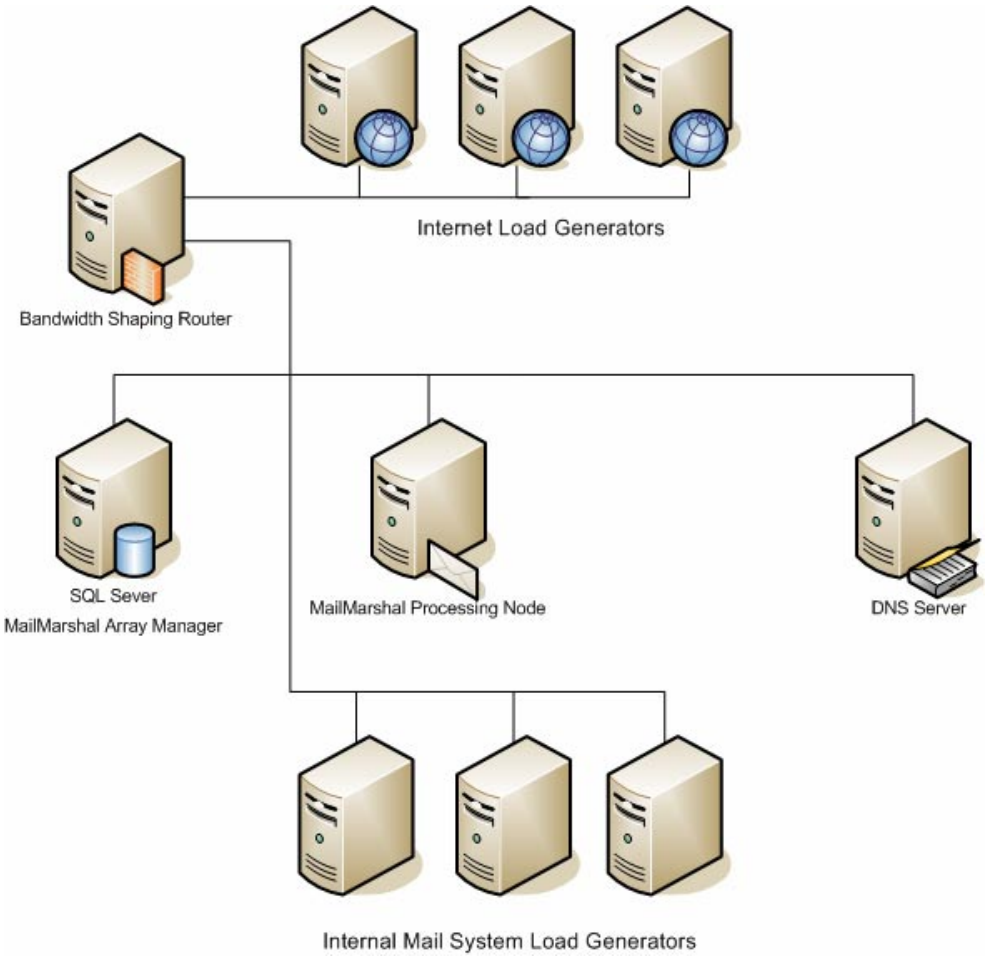
Test Scenarios

Three basic scenarios were tested:

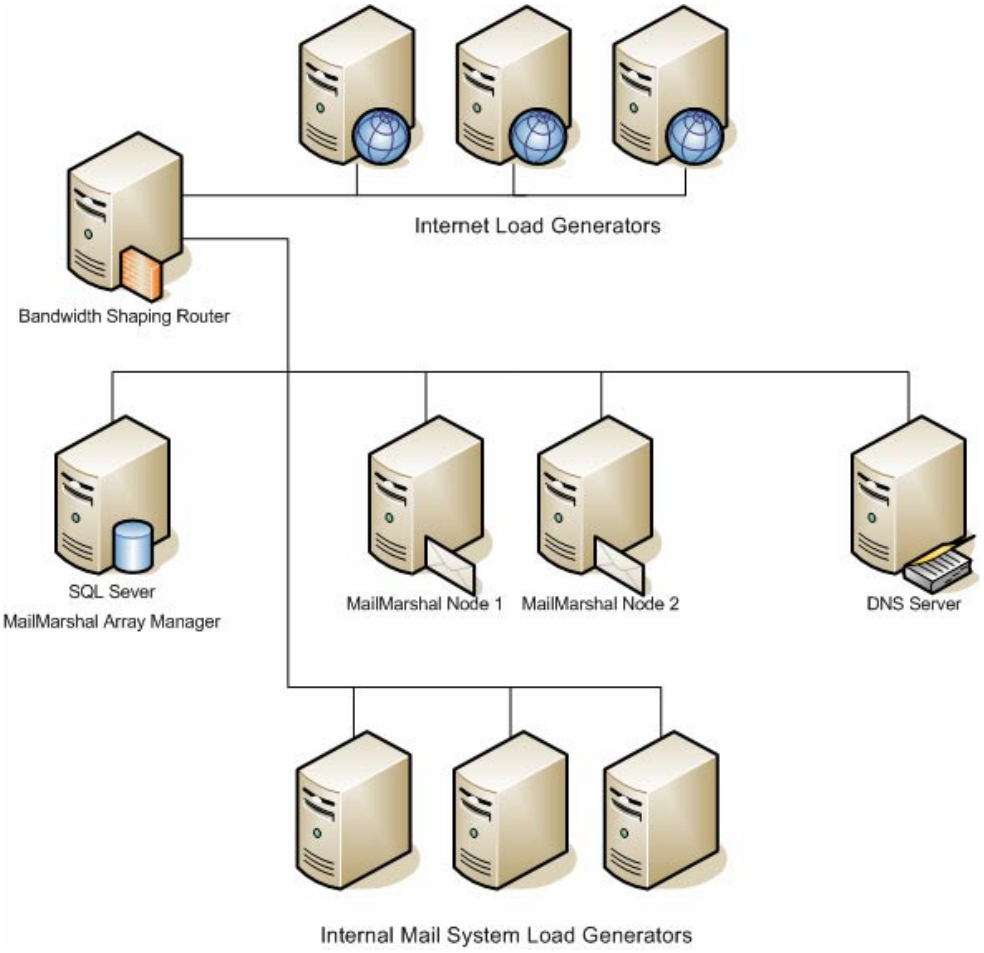
1. MailMarshal running as a standalone server complete with SQL server installed on the same server



2. MailMarshal running as a processing node in an array with the Array Manager and SQL server installed on a separate server.



3. Two individual MailMarshal processing nodes installed as an array with the Array Manager and SQL server installed on a separate server.



Standalone Server

A single Compaq DL360 server was used as the test device, the MailMarshal database was installed onto the "C:" Drive along with MailMarshal itself. The unpacking folder was then moved to the "D:" drive.

The throughput was measured at the MailMarshal 6.0 Engine - figure 5. This showed an average 21 messages/sec for the server. The Bandwidth Shaping was set to unlimited – just act as a router.

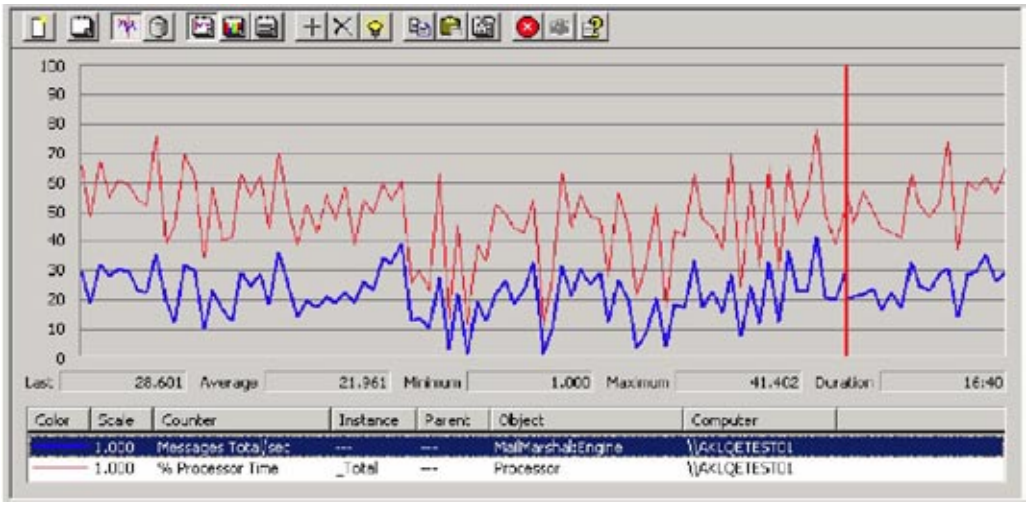


Figure 3 - Standalone server, message throughput in a 16min window, with a 10sec sample rate

Next the bandwidth was restricted to a 250,000 bytes/sec inbound and 250,000 bytes/sec outbound. This is roughly equivalent to a full-duplex E1 (2Mbit) connection.

Again the throughput was measured at the Engine – figure 6. This showed an average of 11 messages/sec.

MARSHAL
Secure. Protect. Comply.

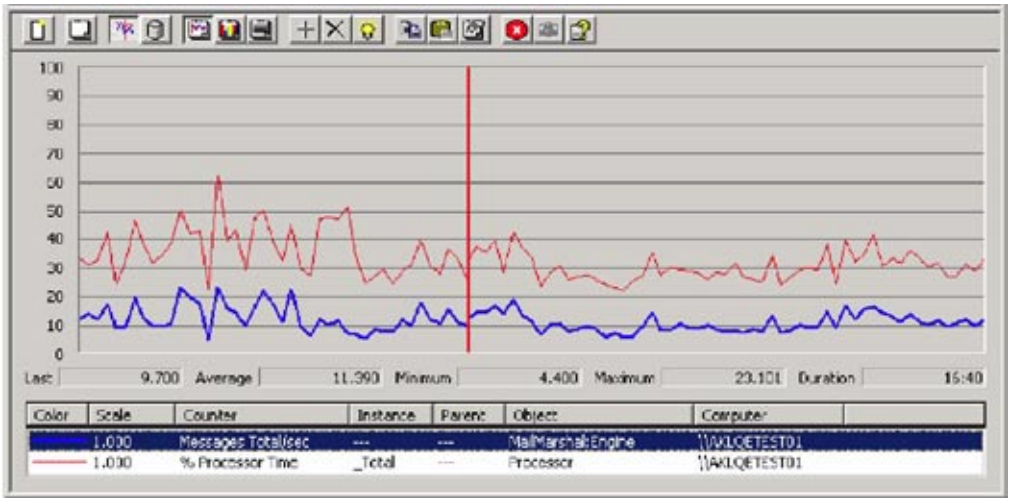


Figure 4 - Standalone server with 2Mbit full duplex link, 16min window with a 10sec sample

Measurement of the bandwidth utilized is shown in figure 7 (Note: the Bandwidth shaping tool was latter confirmed as being somewhat more restrictive than configured), this averaged around 200K bytes/sec in each direction.

The 11 messages per second should yield a bandwidth requirement of 11.3 messages/sec *

32K/message = 361000 bytes/sec, this was consistent with that measured on the Bandwidth Shaping server.

Figure 5 - Standalone server internet bandwidth utilization, 16min window with a 10sec sample

MARSHAL
Secure. Protect. Comply.

Single Node

The results for a single node were predictably half that of the dual node scenario. Processing nodes have no interaction with each other and the coupling to the SQL server is relatively loose so this was expected. See the Dual Nodes scenario for further description and results.

Dual Nodes

Two Compaq DL360 servers were used, each had 2 disks installed and configured as drives C: and D: For both servers MailMarshal was installed onto the C: drive with the unpacking directory set to the D: drive.

Windows Load Balancing was used to spread the load across the two 2 processing nodes.

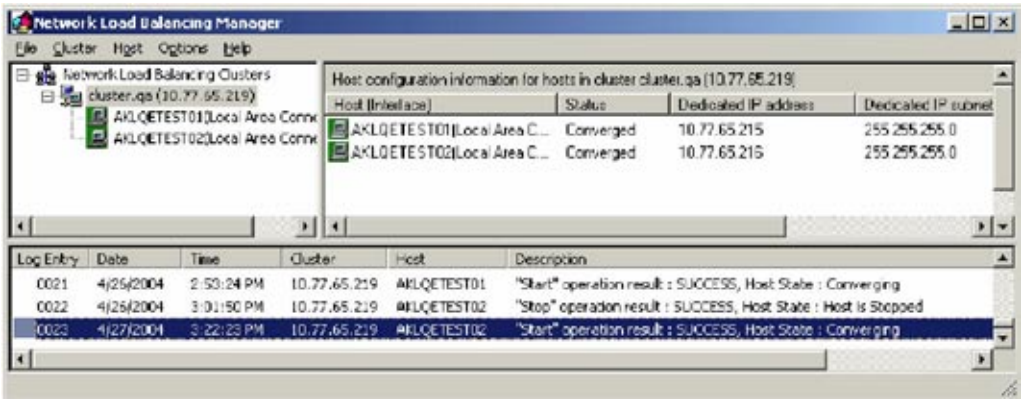


Figure 6 - Dual Node network load balancing

The affinity was set to none so that incoming connections were evenly distributed across the servers. Both the Internal and the Internet Bench-Tool configurations were set to deliver their email to the shared IP address.

The Bandwidth Shaping was set to unlimited – just act as a router.

The throughput was measured at each of the MailMarshal 6 Engines - figure 4. This showed a sustainable 32 messages/sec for each node for a total of 64 messages/sec. This equated to over 2M bytes/sec.

The overall network traffic to the SQL server averaged around 27K bytes/sec with both nodes running, see Figure 5. This low bandwidth requirement coupled with loose connectivity requirements of MailMarshal 6 indicates that the Array Manager / SQL server could potentially be located at an office remote to the Nodes.

MARSHAL
Secure. Protect. Comply.

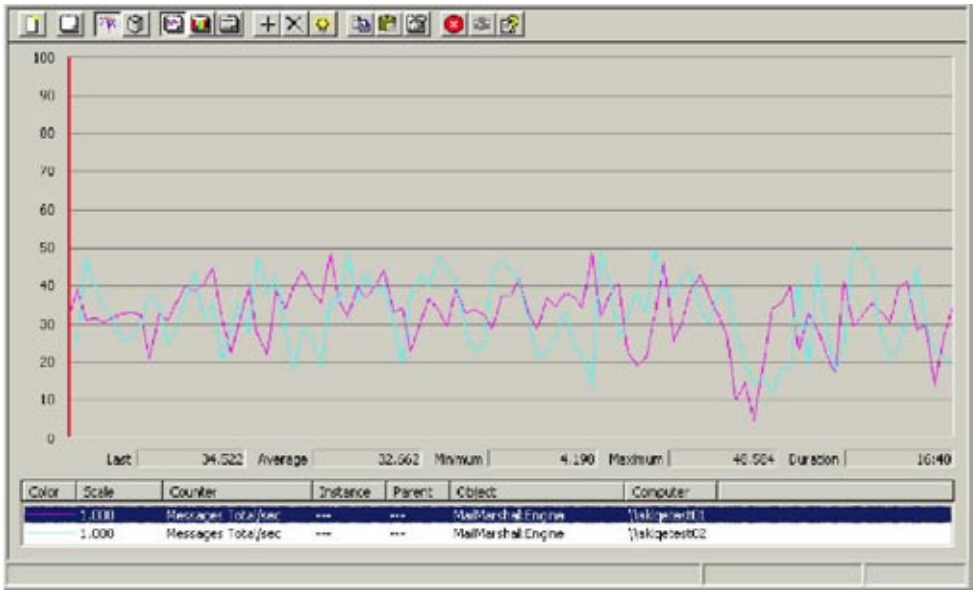


Figure 7 - Dual Node message throughput: 16 min window with a 10 sec sample

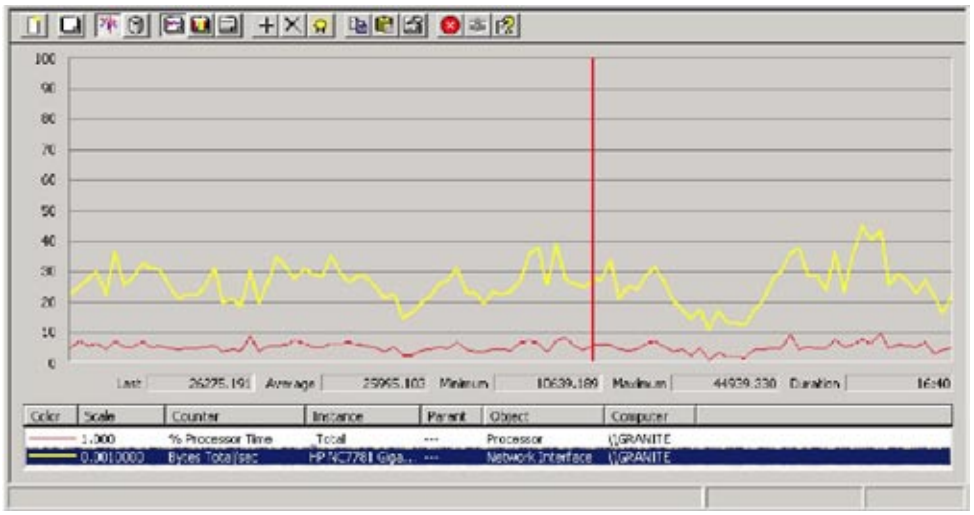


Figure 8 - Dual Node bandwidth to SQL server: 16 min window with a 10 sec sample

MARSHAL
Secure. Protect. Comply.

Summary Discussion

During the initial testing the DL 360 servers were configured so the 2 disks were mirrored – this would be a typical deployment for a 1U rack mount server. It quickly became apparent that disk queue length, which was averaging around 12, was a problem. The servers were reconfigured to provide 2 separate disks and the Unpacking directory was moved onto the second disk.

This promptly increased the message throughput by close to 50% together with making the machine generally more responsive. However this would make a DL360 unsuitable as a node since there is no data protection, a 2U server chassis that supports 4 disks so they could be setup as 2 mirrored pairs would be ideal.

Observation of the Nodes showed both a high CPU usage and high Disk Queue, this indicates a reasonable balance between the I/O and processor performance. Going to a quad CPU machine would not show significant improvement in message throughput unless it was also accompanied by a suitable improvement in disk I/O capability. Strictly from a price performance basis adding more dual processor boxes would be cheaper than going to quads.

The memory footprint of the Nodes was relatively light even under full load, rarely getting above 256MB, however this can vary substantially with some large configurations and large user groups, 1GB of memory would generally be recommended.

The MailMarshal database was created normally which defaults to placing everything in the PRIMARY file group, the "IX_From" and "IX_To" indexes were then moved to a second file group that was on a different physical disk to the PRIMARY file group.

The memory footprint is high – SQL server will take everything it can get; 3GB of memory is generally recommended so SQL can take its maximum of 2GB and still leave some for the Array Manager and the Operating System. Normal operation of the Nodes puts only a light load on the SQL server while using the Consoles and generating reports places a heavier burden on it.

Disk Usage

Apart from the volume of messages that can be quarantined MailMarshal generates significant volumes of diagnostic text logs however the volume of SQL server logging is significantly lighter.

Log sizing:

Receiver	1315 bytes/message
Engine	7710 bytes/message
Sender	1845 bytes/message
SQL	600 bytes/message

By default text logs are deleted once they are over 5 days, SQL logs after 100 days.

Note: The Engine log size is directly related to the number of rules that are defined in the Policy.

The average Spam message size is around 4500 bytes with its associated log file of 4000 bytes, spam represents around 66% of inbound email or 50% of total gateway volume.

Also anecdotal evidence has seen some MailMarshal deployments blocking up to another 50% of inbound email by bandwidth (typically a small number of large messages). Typically most quarantine retention periods are set to around 7 days.

Investigation of several sites yielded an average email volume of 20 messages per person per day (remember this is only email to and from the Internet).

Note: This can vary substantially depending on the nature of the company.

Once spam is removed inbound and outbound mail volumes tend to be similar: Of 20 messages, 10 were spam and 5 were legitimate inbound messages and 5 were legitimate outbound messages. From this information we now have enough data to calculate the disk usage for each user:

- Text logs size is:

20 messages * (1315 + 7710 + 1845) * 5 days retention = 1.087M

- SQL log size is:

20 messages * 600 * 100 days retention = 1.2M

- Quarantine size is:

10 spam messages * (4500 + 4000) * 7 days retention = 595K

- Incoming message size rules

Incoming non-spam volume = 5 * 30K = 150K

150K * 50% * 7days retention = 525K

- Total disk required per user:

Node: 1087 + 595 +525 = 2207K

SQL Server: = 1200K

For example a 100,000 user site will use 340G of disk.

Disk, like most server resources, should not be run near its maximum capacity as the performance can drop quite substantially. Therefore it is generally recommended to add another 50% to ensure optimum performance.

Note: Disk subsystem performance is not only a function of the disk but also of the controller, especially when configuring raid solutions.

Server Sizing

Observation of several Email servers has shown that it is not unusual for 80% of email traffic to arrive in a 10 hour period. The same observation also has shown that the flow of email traffic is fairly linear during that 10 hour window.

Larger sites tend to have more geographically spread offices which can increase this time window substantially, but let's assume our users must send their email in a 10 hour window.

Therefore for 100,000 users we need to ship 100,000 * 20 messages/person * 80% in 10 hours, this is 45 messages/sec.

Based on the test scenarios above, this would require two modern dual processor servers and a separate SQL / Array Manager Server. It would be prudent to then add a third server to ensure that there is no latency in email traffic even in extreme peaks and to provide sufficient capacity should one server fail.

Another measure for server sizing could be done by bandwidth, each Node can deal with a peak of 10Mbit/sec of email, if you only have a 10Mbit/sec link to the Internet then 2 Nodes would be sufficient to give some leeway and cope with a Node failure.

Based on bandwidth, a full-duplex E1 connection should support 2M / 10bits/byte / 30K = 6.6 messages/sec inbound typically this would mean 3.3 messages/sec outbound for a total of 10 messages/sec.

Relating this back to an average user count we get 10msgs/sec * 10hours * 60mins/ hour * 60secs/min/ (20msgs/user * 80% in 10hours) = 22,500 users.

Although the test scenarios showed that a MailMarshal processing node could sustain a rate of 32 messages/sec, it would generally be unwise to design a system where the expected load per node averaged above 20 messages/sec. A connectivity fault could result in large volumes of email being held and then delivered to MailMarshal in a hurry; at that point the extra capacity will be desirable.

MARSHAL
Secure. Protect. Comply.

Summary

It is readily possible to build MailMarshal environments that will support well in excess of 100,000 users. Over the period of the week spent testing the various scenarios MailMarshal processed over 15 million messages without fault.

Key findings, based on 20 messages per user per day with an average size of 30K where 80% of these are sent in a 10 hour window:

• 1 standalone MailMarshal can sustain 20msgs/sec. This is easily sufficient to cope with a dedicated E1 or T1 email connection or around 22,500 users.

• 1 MailMarshal processing node with a separate Array Manager can sustain 32 messages/sec. This is the generally recommended deployment option for MailMarshal as reporting and console actions do not impact node throughput. This would be sufficient for a dual E1 connection or around 45,000 users.

• 2 MailMarshal Nodes with separate Array Manager can sustain 64 messages/sec.This solution would be sufficient to cope with up to 90,000 users.

Even under high load the console was responsive and messages could easily be located and appropriate actions taken.

MARSHAL™
Secure. Protect. Comply.

Appendix

Bench Tool Description

The Bench-Tool is proprietary software developed by Marshal specifically for performing sustained load tests.

The Bench-Tool both submits email to the server under test and accepts email from that server. It reads an rfc822 email message from a source directory, re-writes the "To:" and "From:" fields with a configurable range of random users and then sends the message to the server under test. The messages in the source directory are read in alphabetical order, restarting at the beginning when all have been sent.

In order to regulate the flow of email the Bench-Tool employs a feedback loop; each message the Bench-Tool sends has a header line inserted that has the originating machine name and an incrementing sequence number. When a message arrives at an instance of Bench-Tool, the sequence number is advertised to the originating instance of Bench-Tool, this allows the Bench-Tool to limit the number of messages being sent in to the rate at which they can be processed and delivered.

Typically an instance of the Bench-Tool is configured to allow the last advertised message to be up to 100 messages behind the sequence number of the current message it is trying to send. Generally the system works well; it is not sensitive to messages which may be quarantined (provided there isn't a run of more 100 messages quarantined). It is a little vulnerable to messages that get delivered out of order as the Bench-Tool then believes it can submit more messages to be processed – this just results in more than 100 messages being outstanding. In the test scenarios described above we ran 6 copies of Bench-Tool resulting in 600 messages in transit at any one time, when testing with 2 nodes we eventually increased this to 200 messages per instance as occasionally the Engine would stall having completely emptied the incoming queue.

The Bench-Tool utilises a MailMarshal 5.5 Receiver for accepting email and a MailMarshal 5.5 Sender for submitting email and is therefore fully multi-threaded.

MARSHAL
Secure. Protect. Comply.

MailMarshal 5.5 Comparison

MailMarshal SMTP 5.5 was installed onto one of the Nodes and its default configuration was used, Sophos Anti-Virus enabled, the unpacking directory was set to the second disk to spread disk I/O, logging was directed to a separate SQL server.

MailMarshal SMTP 6.0 clearly has improved on the message throughput capabilities of MailMarshal SMTP 6.0.

MailMarshal SMTP 6.0 (32.66 messages/sec) is approximately 40% quicker than MailMarshal 5.5 SMTP (23.45 messages/sec) while offering a significantly improved console functionality.

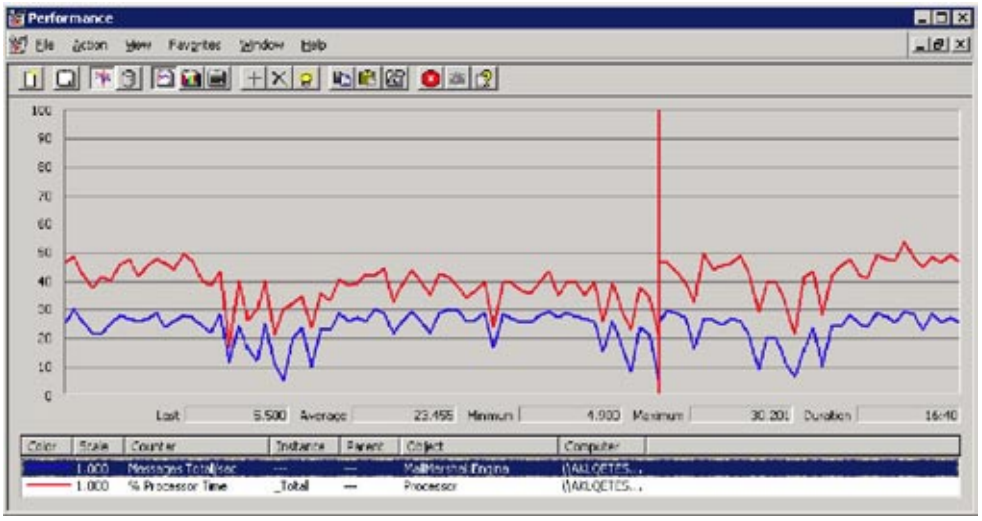


Figure 9 - MailMarshal 5.5 message throughput: 16 min window with a 10 sec sample

Marshal's Worldwide and EMEA HQ
Marshal Limited,
Renaissance 2200,
Basing View,
Basingstoke,
Hampshire RG21 4EQ
United Kingdom

Phone: +44 (0) 1256 848080
Fax: +44 (0) 1256 848060

Email: emea.sales@marshal.com

Americas
Marshal, Inc.
5909 Peachtree-Dunwoody Rd
Suite 770
Atlanta
GA 30328
USA

Phone: +1 404-564-5800
Fax: +1 404-564-5801

Email: americas.sales@marshal.com

Asia-Pacific
Marshal Software (NZ) Ltd
Suite 1, Level 1, Building C
Millennium Centre
600 Great South Road
Greenlane, Auckland
New Zealand

Phone: +64 9 984 5700
Fax: +64 9 984 5720

Email: apac.sales@marshal.com

info@marshal.com | www.marshal.com