



MailEnable Scalability White Paper

Version 1.2

Table of Contents

1	Overview.....	2
2	Core architecture.....	3
2.1	Configuration repository	3
2.2	Storage repository	3
2.3	Connectors	3
2.3.1	SMTP Connector	3
2.4	Agents	4
2.4.1	Mail Transfer Agent	4
2.5	Services	4
2.5.1	POP3 Service.....	4
2.5.2	HTTPMail Service.....	4
2.5.3	Web mail service	5
3	Performance Tests.....	6
3.1	Minimal Hardware Configuration.....	6
3.1.1	Incoming Mail	6
3.1.2	Outgoing Mail.....	7

1 Overview

This white paper outlines scalability considerations for the various MailEnable mail server products.

This document only attempts to outline negative scalability “issues” and therefore may appear quite critical of the MailEnable product. *(We figure that if anyone can afford to be critical of the product’s implementation; we can!)*

This document is limited only to the services outlined within it. (For example, the List Server and POP Retrieval Connectors are not analysed or explained in this document).

It is recommended that you read the MailEnable System Manual (which is installed with MailEnable) as background to this document.

2 Core architecture

MailEnable is comprised of two principal repositories (storage groups); the configuration repository and the actual message store.

The configuration repository stores MailEnable's meta-data (eg: mailbox names, post office names, domain names, address mappings, etc). In the base or standard installation of MailEnable, this information is stored in tab-delimited files.

The second principal repository used by MailEnable, the storage repository, is used to store mail messages, folders and queues. These are stored under a logical directory structure on a file service.

2.1 Configuration repository

MailEnable Standard and Professional Editions incorporate tab-delimited configuration providers. These tab delimited providers somewhat limit MailEnable's ability to scale, primarily because of searching and updating these tab delimited files. On modest hardware, this limitation has an impact at around 40,000 subscribers.

The limitations associated with MailEnable's configuration repository (using the tab-delimited configuration provider) are explained later in this document under the headings of the connectors, services and agents that use them.

Note: MailEnable Enterprise Edition allows the use of databases for storage of configuration data and therefore overcomes the limitations associated with using tab-delimited configuration providers).

2.2 Storage repository

MailEnable stores messages as uncompressed text files. The key advantage of doing this is that it facilitates clustering and is simple to backup and restore.

As messages are stored uncompressed, MailEnable can access them very quickly. However, with fast access, there is a trade off in terms of disk consumption.

For example, assuming 40,000 subscribers with 10 MB storage limits could occupy around 400,000 MB of storage (or 400 GB). In normal ISP terms though, the mean consumption is around 60% of capacity, hence 240 GB is likely to satisfy the load.

MailEnable's storage repository is not indexed and therefore increased CPU load may be experienced where there are mailboxes with significant numbers of messages. The Enterprise version produces an index file for each mailbox folder to speed message access. This method is also used for IMAP in the Professional and Enterprise Editions.

There are also disk fragmentation issues where large numbers of messages are being moved through connectors. The operating system provides some relief here.

2.3 Connectors

2.3.1 SMTP Connector

The SMTP Connector is responsible for receiving and sending mail via the SMTP protocol. Most mail received by the MailEnable will arrive from mail clients or other SMTP servers via the SMTP Connector. The SMTP Connector is also used to dispatch mail to remote hosts (sending mail).

2.3.1.1 SMTP Inbound

Inbound SMTP message throughput is primarily limited by bandwidth availability. MailEnable can perform a reverse lookup on the IP of those hosts sending inbound SMTP mail.

Note: When running under Windows NT the DNS lookup is single threaded and will impact on performance.

The SMTP Connector can optionally resolve recipient addresses as they are passed via the RCPT command. If MailEnable is servicing a large number of addresses, traversing these addresses will increase CPU utilization (if using the tab-delimited configuration provider).

In most cases, SMTP Authentication is required for relaying and therefore, inbound SMTP clients will need to authenticate against the Authentication Provider (which is a tab-delimited provider). Because this only occurs each time a “trusted” user wishes to dispatch an outbound message (relay), this is not a major scalability issue.

2.4 Agents

2.4.1 Mail Transfer Agent

The Mail Transfer Agent (MTA) is extremely efficient at moving messages between connectors (because it is a simple task). The MTA uses an “Address Map” table to determine how to move mail between connectors. Each mail address served by MailEnable will occupy an entry in the Address Map table.

For example: An entry would be required in the Address Map Table to indicate that mail addressed to [SMTP:Peter@mailenable.com.au] should be redirected to the [SF:MailEnable/Peter] Postoffice/Mailbox

Standard Edition provides this table as a tab delimited file (ADDRESS-MAP.TAB). There are some scalability issues associated with using this because in most cases, the entire table needs to be parsed in order to transfer a message between connectors. This may become an issue where MailEnable is processing more than 40,000 addresses (if using the tab delimited provider).

Note: An alternative is to use the database connector available in MailEnable Enterprise Edition.

2.5 Services

2.5.1 POP3 Service

The POP3 service supplied with MailEnable Standard Edition is configured to use tab delimited files. The primary scalability consideration for the POP3 connector is the actual authentication process. If the AUTH.TAB file contains more than 40,000 logins, POP authentication may be CPU intensive (as it must parse the AUTH.TAB file to validate the login). Fortunately, this occurs only once per POP3 session and the likelihood of multiple concurrent requests is low.

2.5.2 HTTPMail Service

The HTTPMail Service accesses the repository. HTTPMail is a connectionless protocol and is therefore scalable.

The HTTPMail service, by default, is configured to use tab delimited files. The primary scalability consideration for the HTTPMail service is the actual authentication process. If the AUTH.TAB file contains more than 40,000 logins, authentication may be CPU intensive (as it must parse the AUTH.TAB file to validate the login).

Fortunately, MailEnable caches up to the last 100 accessed credentials and only needs to access this file when an un-cached login attempt is made or where MailEnable detects that the cached credentials do not match those supplied by the client application.

Using the tab delimited configuration provider with HTTPMail should not pose any significant scalability issues. MailEnable Enterprise Edition facilitates using a database to store credentials, and is therefore much more scalable.

2.5.3 Web mail service

2.5.3.1 Loading message list

When a list of messages is requested by the Professional Edition web mail service, the entire directory /Folder needs to be scanned and each message file opened to interpret the headers. If a folder/directory contains thousands of files, returning the list of messages within the folder will be CPU and Disk IO intensive.

The ideal situation here is to limit the number of messages that are held in the users inbox.

The Enterprise Edition web mail uses index files to store summary information about each message, cutting down disk access to reading one file per folder and provided a large performance benefit for web mail usage.

2.5.3.2 Uploading Attachments

Uploading and downloading of attachments require base-64 encoding and decoding routines. Large attachments or large numbers of attachments being accessed by web mail will elevate CPU utilization.

3 Performance Tests

3.1 Minimal Hardware Configuration

The following case study outlines a load test on minimal hardware. The oldest available hardware was used.

Server	Network	Client Simulator
Clone PC Single CPU Athlon 1 GHz 256 MB RAM Windows 2000 Server	10bT Ethernet Hub	HP OmniBook XE/2 PC Notebook Single CPU Pentium II 400 MHz 196 MB RAM Windows 2000 Server

Note: In all cases, the default installation parameters have been used. (Tuning the server could significantly improve the results).

3.1.1 Incoming Mail

The test measured the rate at which the SMTP Connector accepted mail from remote clients.

The client simulator ran 20 instances of a Test Harness utility that sent messages to the SMTP Connector. (These tests can be replicated using tools like Microsoft's ESP Utility). The actual load test application is not important; the intention is to hit the server with as many connections as possible. To remove any bias, only the number of messages actually received by the SMTP Connector was used.

The actual message transferred from the client to the server was a 2K MIME encoded message containing the EICAR virus. (Virus checking was disabled on the MTA). The test period ran over 60 continuous minutes.

Messages were received by the SMTP Connector and placed in the inbound message queue. The MTA performed an Address-Map lookup and moved the messages to the Post Office Connector Outbound Queue. The Post Office Connector then processed the messages and delivered them to the appropriate mailbox.

At the completion of the test period all MailEnable Connectors, Services and Agents were stopped using the Service Control Manager. Once all services were stopped the following message quantities were left in their respective locations:

TABLE 1: MESSAGES QUEUED/PROCESSED

Location	Number of Messages
SMTP Connector Inbound Queue:	0
Post Office Connector Outbound Queue	5,275
Post Office Mailbox	17,806

Using this information, it is possible to determine the number of messages (per second) that are moved through MailEnable services.

This is outlined in the following table:

TABLE 2: MESSAGES (PER SECOND) BY SERVICE

Indicator	Duration	Total	Messages Per Second
Total SMTP Messages Received	60 Minutes	23,081	6.5
Total MTA Messages Transferred	60 Minutes	23,081	6.5
Total Post Office Messages Delivered	60 Minutes	17,806	4.9

3.1.2 Outgoing Mail

This test indicates the rate at which MailEnable can dispatch messages to remote hosts. As mentioned earlier, the major limiting factor associated with dispatching messages to remote domains is the actual DNS lookup on the domain to determine which host should receive the message.

The current product has been load tested by independent users of MailEnable and can deliver around 20,000 newsletters over a 3-hour period (of course this figure depends on the responsiveness of the DNS Server being used and the link quality). Assuming these metrics, MailEnable will dispatch at least 2 messages per second.