# Mirapoint RazorGate 300 V3.5.4
## Technical Evaluation

**An NSS Group Report**

First published January 2005 (Version 1.0)

Published by The NSS Group
Security Testing Laboratories
Mas la Carrière, Route de Ganges
30440 Sumène, France

Tel : +33 (0)4 67 81 49 11
E-mail : info@nss.co.uk
Internet : http://www.nss.co.uk

# TABLE OF CONTENTS



# TABLE OF FIGURES

# The NSS Group

The NSS Group is the world's foremost independent security testing facility.

With British headquarters, and security and network infrastructure testing facilities in the South of France, The NSS Group offers a range of specialist IT, networking and security-related services to vendors and end-user organisations world-wide.

**The NSS Group's Security Testing Laboratories** are available to vendors and end-users for fully independent testing of networking, communications and security hardware and software.

The NSS Group also operates certification schemes for vendors and certification bodies, and currently provides evaluation and certification of a wide range of security products, including IDS/IPS appliances, firewalls, VPN's, Web Application firewalls, multi-function security appliances, cryptographic devices and PKI products.

Output from the labs, including detailed research reports, articles and white papers on the latest network and security technologies, are made available on the NSS web site at http://www.nss.co.uk.

The NSS Group awards are recognised world-wide as being the most desirable and essential when it comes to security products. Vendors consider the awards to be a crucial step in any security-related marketing campaign, whilst feedback from readers of the reports indicates that participation in an NSS Group test and/or one of the **NSS Approved** awards is a prerequisite for any security product in order to be considered for purchase.

# INTRODUCTION

Almost from day one it has been apparent that e-mail is one of ***the*** killer applications for the Internet. One of the major reasons many of today's businesses get on line is to provide e-mail for their employees. Within a very short space of time, those companies are relying on e-mail communications as a core part of their dealings with both customers and business partners.

*Reliable* e-mail is a must. *Secure* e-mail is a must. *Manageable* e-mail is a must.

Undoubtedly, electronic communication is a vital asset to the modern business, even the life blood of some, and when things go wrong and the mail server is down for hours – or even days – then operational efficiency can take a serious nose dive. Unfortunately, most corporate e-mail systems start life in a small way, with just a few users, and grow as the potential is recognised. They are rarely designed from the ground up with hundreds or thousands of users in mind, and this can eventually result in performance problems and security risks.

It has become all too easy, for instance, to unthinkingly transmit sensitive corporate information over an insecure channel – the Internet – without giving it a second thought. Just click on the "attach" button and away it goes. Such transmissions need to be secured effectively.

Likewise, we are so used to receiving such attachments that we have become susceptible to receiving viruses embedded within otherwise innocent-looking documents. Such transmissions need to be stopped "at the door", before reaching the user's mailbox.

On a more mundane level, senior management needs to take an active interest in what sort of material in general is being transmitted via corporate e-mail systems. Even personal views of individual employees can land a company in legal hot water should it contain anything defamatory or libellous, so it is essential that a clearly defined e-mail policy is in place before letting employees loose on the Internet. This policy should clearly spell out what is and is not allowed, and the penalties for transgression. But then, of course, there is the problem of enforcing that policy!

## E-Mail Security Issues

Over 450 new viruses are discovered each month, according to IDC Research. Gartner Group estimates that more than 80 per cent of computer viruses enter the network through e-mail, and the typical infection costs organisations up to $500,000 per incident.

As mentioned earlier, e-mail has become such a common medium for transmitting files that we are used to seeing attachments with our e-mails. When configured in a certain way (which used to be the default out of the box, and so is probably still the current configuration for many thousands of computer users around the world), Windows hides file extensions of file types that it recognises, such as .EXE (program files), .DLL (libraries), .VBS (Visual Basic Scripts), and so on. Thus, it is possible to craft a VB Script with a nefarious payload and call it something like NAKED.JPG.VBS and this script can then be sent as a normal e-mail attachment.

Because Windows hides the file extension by default, the file name appears as NAKED.JPG. The user, thinking this is simply an image file, double-clicks on it to view and the script is activated. Clearly similar methods can be used to transmit other types of Trojan and virus files.

One way around this threat is to discard all e-mail attachments as they enter (or leave, assuming you also wish to stop your own employees from propagating viruses) the corporate network. Unfortunately, whilst this may solve the virus problem to a certain extent, it drastically reduces the usefulness of e-mail. Attachments have become a common and widely accepted method of transmitting information, with the bulk of the data carried within an attached Word document or Excel spreadsheet, and the e-mail itself consisting of a one-liner saying little more than “read this”.

Relying on every user to ensure that their desktop AV scanner (if they even have one!) is up to date is simply not viable in certain situations (universities with thousands of student laptops over which they have no direct control, for example), and so it is essential to perform affective AV scanning on, or adjacent to, the mail server. If mail traffic is scanned as it actually enters or leaves the corporate network, there is less chance of allowing viruses through to the end user.

Whilst Anti Virus scanning has migrated from the desktop to the gateway server in the last few years, AV scanning alone is not longer enough. Many corporate users see spam as almost as big a threat as viruses these days - whilst not as destructive, it can cost as much, if not more, in lost productivity. The average worker receives more than 13 spam messages a day, claims Nucleus Research, which requires six and a half minutes to process, thus reducing employee productivity. According to META Group, 5-15 per cent of corporate e-mail is spam, and this is expected to grow to 15-30 per cent in the near future. The Radicati Group projects the world-wide cost of spam is $20.5 billion per year.

Spam-related threats include open relays that can be exploited by unscrupulous senders to route large volumes of spam through an organisation’s message network without their permission or awareness. For the organisation whose mail server is used to perform the relay of the spam, the consequences can be dire. At best, server and Internet bandwidth resources can be consumed for hours or days on end. At worst, the otherwise innocent organisation’s ISP could decide that connectivity should be terminated.

Yet another threat is directory harvesting, in which spammers employ public or known e-mail addresses to steal other valid e-mail addresses from a corporate or service provider mail server, and either sell them to other spammers or use the lists themselves.

At best, spam is an irritation as it floods mail boxes with unwanted messages, and at worst it can fill a user’s mailbox or overwhelm an e-mail server, effectively blocking legitimate mail. Inbound spam can be partially controlled by keyword-searching inbound e-mails and rejecting likely messages. This technique should be used carefully, however, as it can easily lead to incorrectly rejecting valid e-mails.

Solutions to this problem have become more and more advanced, with complex lexical analysis engines applying “fuzzy logic” to scanning e-mails for content which could be considered spam.

Of course, the problem with spam is that it is even easier for a spammer to change the content of his message - even on a per-e-mail basis if he wants - than it is for the virus author to change his nefarious payload. Thus it is simply not possible to create accurate “signatures” to identify spam. Instead, a multi-layered approach should be taken, including the use of the aforementioned lexical analysis, external real-time blacklists, validation of sender information, and so on.

Beyond viruses and spam are even more menacing security threats. Denial of Service (DoS) attacks, sometimes referred to as mail bombs, are designed to bring a network to its knees by flooding it with useless traffic. Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), and Internet Message Access Protocol (IMAP) ports are subject to the same DoS attacks that impact Web servers. While there are software fixes available to limit the damage caused by the attacks, like viruses, new DoS attacks are continually being created by hackers.

And then, there is the issue of enforcing a strict corporate e-mail policy. Businesses are faced with an increasing number of regulations governing e-mail traffic, requiring vertical industries like finance and healthcare to devise their own requirements to document transactions, archive communications, protect privacy, and ensure honest business practices.

Examples include the *Gramm-Leach-Bliley Act*, the *Sarbanes-Oxley Act*, and the *Health Insurance Portability and Accountability Act (HIPAA)* of 1996. Enterprises covered by these regulations must either comply or face possible civil and, in some situations, criminal liability.

## Enforcing E-Mail Security Policy

Enforcement of e-mail security policy can only be really effective when applied at the e-mail gateway to the Internet or on the backbone, not at the user’s desktop. It is not practical, or financially viable, to replace legacy systems or to implement enhancements on every desktop. The solutions that are required must add value to the existing e-mail system, whilst being implemented and controlled at a single point where e-mail enters and leaves the organisation – the boundary.

Traditional messaging solutions may not be able to effectively address these security issues. Old-world, software-only messaging solutions were created in an era before security was a problem. E-mail security solutions are typically assemblies of point product solutions (such as general-purpose servers, virus scanning, and anti-spam applications) integrating hardware and software from multiple vendors that were neither specifically created nor optimised to work together. Correctly configuring these disparate systems or applications can be complicated and error prone, exposing businesses to security threats. Dedicated e-mail appliances and dedicated e-mail security gateway devices can go a long way towards making lifer easier and more secure for the beleaguered administrator.

Managed correctly, e-mail can be a considerable asset to any organisation. It can improve communication between employees, between departments within a company, between geographically dispersed offices, and between customers and business partners. In order to maximise the beneficial effects of e-mail, however, it is necessary to implement is carefully and select a mail system that is scaleable, flexible, manageable, robust and secure.

# MIRAPOINT RAZORGATE 300 V3.5.4

## Architecture

The architecture of the RazorGate 300 is very simple, given that it is a stand-alone appliance designed to provide a range of e-mail filtering and security features in a single box. It runs a customised and hardened operating system called MOS (*Messaging Operating System*) which is Unix-based and runs only those services necessary to operate as a mail-filtering appliance.

RazorGate is not, in itself, a mail server - it is designed instead work in conjunction with existing corporate mail servers. Since it conforms to all mail-related standards, it is vendor-agnostic and will work with any existing mail server, not just those from Mirapoint.

RazorGate appliances combine Anti Virus, Anti Spam and Content filtering capabilities into a single, dedicated, integrated appliance. Whilst normally acting as a full mail proxy in a store-and-forward mode, the optional DirectPath Scanning feature enables the device to work in a queue-less environment.

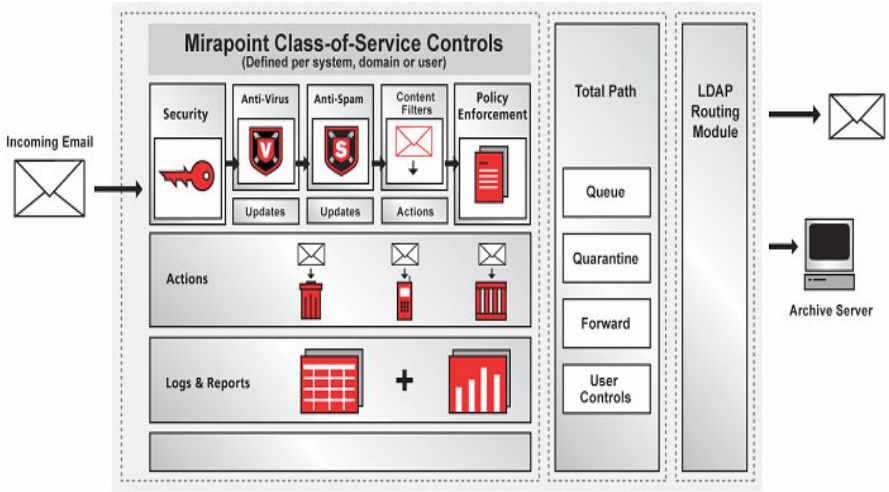


*Figure 1 - RazorGate: Architecture*

With DirectPath, mail is not stored locally on the appliance. Instead of running as a proxy, RazorGate opens a direct connection from sending to receiving mail server, and filters mail in real-time as it passes through. DirectPath thus allows real-time scanning without having to worry about queue management or storage and backup of mail in a queue.

In addition, the message transfer agent (MTA) guarantees no inbound messages are lost. The MTA does not allow a message to be acknowledged as being accepted by the RazorGate appliance until the message has been fully delivered and acknowledged by the recipient mail server.

This concept is similar to a two-phase commit process in the database world. If a RazorGate appliance were to lose power, any message that was in process within the appliance would never have been acknowledged as being accepted, and delivery would be re-attempted by the sending server using standard SMTP behaviour.

Whilst this may be attractive to users of the RazorGate 100 which has no hardware redundancy or battery-backed cache (and thus no way to protect mail stored temporarily in local queues), we feel that most environments would prefer the added security of the full mail proxy offered by the standard RazorGate deployment. At least the option is available.

When DirectPath is not deployed, RazorGate provides extensive queue management capabilities. In addition, a quarantine feature enables the administrator to review suspicious mail manually before deleting or forwarding, and an extensive range of monitoring, auditing and logging tools are included. Three versions of the RazorGate product are currently available:

| | |
|---|---|
| **RazorGate 100** | ■ 1U rack mount appliance<br>■ Xeon processor<br>■ Bundled with 100 users<br>■ Designed for customers with <1500 users |
| **RazorGate 300** | ■ 3U rack mount appliance<br>■ Xeon processor<br>■ Hardware redundancy<br>■ Battery-backed RAID disk controller<br>■ Hardware watchdog provides auto-recovery<br>■ Bundled with 500 users<br>■ Designed for customers with 1500 to 5000+ users |
| **RazorGate 450** | ■ 3U rack mount appliance<br>■ Dual Xeon processors, additional memory increased performance<br>■ Hardware redundancy<br>■ Battery-backed RAID disk controller<br>■ Hardware watchdog provides auto-recovery<br>■ Multi-system management capability<br>■ Unbundled (any number of users)<br>■ Designed for customers with 5000+ users |

The unit provided for testing was the RazorGate 300, running MOS V3.5.4. This is a 3U rack mount appliance with a single 2.80 GHz Intel Xeon processor and 1GB of ECC DDR SDRAM. Accessible behind the front panel are six drive bays, three of which are occupied by default with three 10,000RPM SCSI hard drives providing a total of 53 GB RAID-1 storage (two RAID-1drives plus one hot spare). The integrated Ultra160 high performance SCSI RAID controller has 64 MB of cache memory and battery backup.

Thanks to the tight integration of the hardware and software in this appliance, the RAID controller and disk storage are fully manageable from the RazorGate GUI, meaning that device health is visible and RAID drives can be rebuilt from the normal management interface without having to resort to using the drive manufacturer's utilities.

Also on the front panel is a large LCD display with menu buttons, providing straightforward access for the administrator to basic management functions such as setting the initial IP address of the unit, reboot, shutdown, and so on.

On the rear panel are two copper Ethernet ports - one 10/100/1000 Gigabit port, and one 10/100 Fast Ethernet port. Note that this is not an in-line device, and so only one of these ports will be connected to the corporate network for mail processing.

Next to the Ethernet ports are the twin redundant hot-swappable power supplies. Should one of these fail, the device will keep on running, but an alarm will sound, and e-mail alerts will be sent to the administrator (plus SNMP traps to the network management system if configured). Removing and replacing the power supplies is a simple job, taking only moments.

The four cooling fans inside the unit are also hot swappable. The appliance slides out on its rails to the halfway point where the fans can easily be removed by undoing two thumb-screws. As with the power supplies, fan failure generates e-mail alerts and SNMP traps (if configured).

With any hardware failure, e-mail alerts are also sent to Mirapoint Customer Care, and as part of the Mirapoint support contract replacement parts are shipped overnight to the customer. In some cases, the replacement parts could arrive before the administrator is even aware of a failure! High availability and redundancy has been addressed thoroughly in the RazorGate 300.

Mirapoint MOS can be updated easily in one of two ways, either via the GUI or CLI. When using the CLI, the update is a one line command which is performed manually. When using the GUI, however, the administrator can set automated notification of new updates, along with recommendations on need/severity of the update (strongly recommended, recommended, advisable or optional). The available updates are shown in the GUI, along with the recommendation and a one click button to install the update.

## Installation

As with any mail filtering system, the key to a successful installation is the correct configuration of DNS and firewalls. Whilst the RazorGate product does not demand anything special in the way of DNS/firewall configuration, nevertheless this is usually the cause of any initial problems the administrator is likely to face.

Compared with the potential difficulties of configuring DNS, installation of the device itself is incredibly simple. Once the IP address has been entered via the front-panel LCD, configuration of the RazorGate is via a straightforward browser-based GUI. The administrator can configure the GUI to be accessible over HTTP or HTTPS as required, and can also restrict access to specific IP addresses. Given that there are two network interfaces in this device, one of them can be used as a dedicated management connection if required.

There is also an optional Operations Console available for managing multiple devices in large-scale deployments, and for those who are more at home with the command line than a GUI, there is a rich command line interface (CLI). This provides all the features of the GUI and more, since there are a large number of commands which were simply too complex to represent in a browser-based console, and is accessible via plain text (telnet) or secure (SSH) channels (we would prefer to see SSH as the default, however).

Finally, there is a complete API available (the existing GUI is written using it) which is documented in minute and extensive detail in a separate manual, and which provides the ability for customers to write their own applications which interface directly with the RazorGate appliance.

Depending on the application of the RazorGate, the system can be configured for inbound routing only (accepting mail from the Internet, filtering, and passing to an internal mail store), outbound routing only (accepting e-mail from an internal mail store, filtering, and passing to the Internet), or both. Mail routing is generally performed on a per-domain basis, but can be overridden on a per-user basis by the use of distribution lists or by using a per-user entry in the routing table.

Note that by default, even when configured to handle both inbound and outbound routing, spam filtering is performed on **inbound** traffic only. There is a workaround for this which entails execution of a single command at the command line interface which ensures that spam filtering is performed on both inbound and outbound traffic - it would be preferable to provide such a straightforward option via the GUI, however.

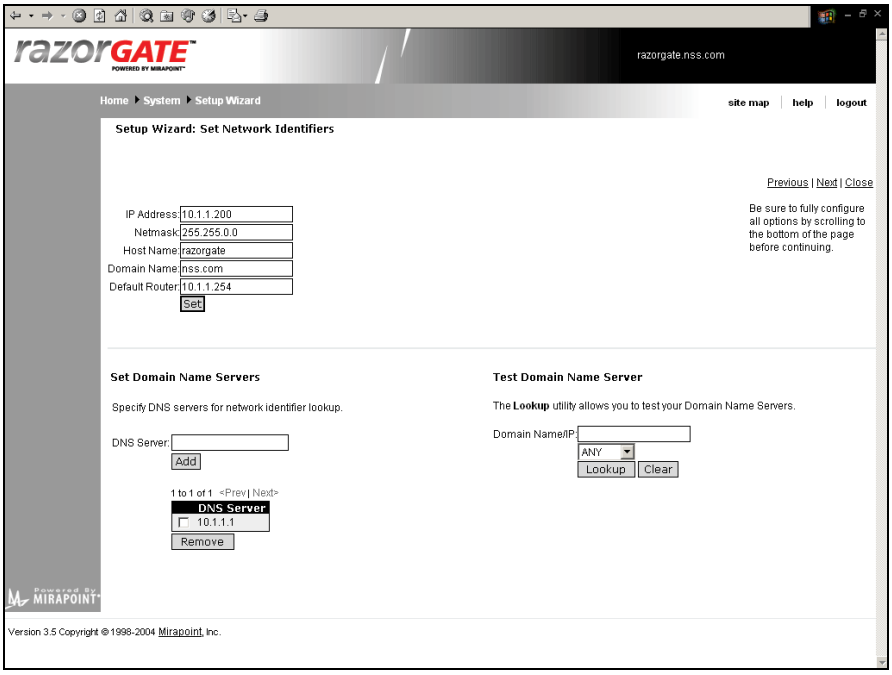


*Figure 2 - RazorGate: The RazorGate Setup Wizard*

Where the GUI does excel is with the *Setup Wizard*, which is activated automatically the first time the device is accessed following installation. This walks the administrator through all the necessary steps required to configure the device, and provides excellent on-screen help for most of the options, explaining all of the available choices in enough detail that recourse to the extensive and excellent administration manuals is not necessary.

Here, the administrator gets to specify the date and time, network settings, license keys, mail domain(s) which will be serviced by RazorGate, outbound Relay List (ensuring that RazorGate itself cannot be used as an open relay), routing method (local routing or external LDAP servers can be used) and recipients of alert messages.

He also gets to enable or disable the various services running on the RazorGate appliance, such as SMTP, SNMP, and the IMAP, POP and HTTP proxies.

Once configured, the Setup Wizard can be run again as many times as required, or individual settings can be configured via the various menu options. Each area of the system (*System Configuration, Anti Virus, Anti Spam, Content Filtering, Queue Management, Security Settings, Monitoring, Logs & Reports* and *Performance Graphs*) are accessible via their own menu options in a menu bar down the left of the screen. However, for those who get lost easily, a complete site map is also available, setting out every menu option in a hierarchical menu on a single screen.

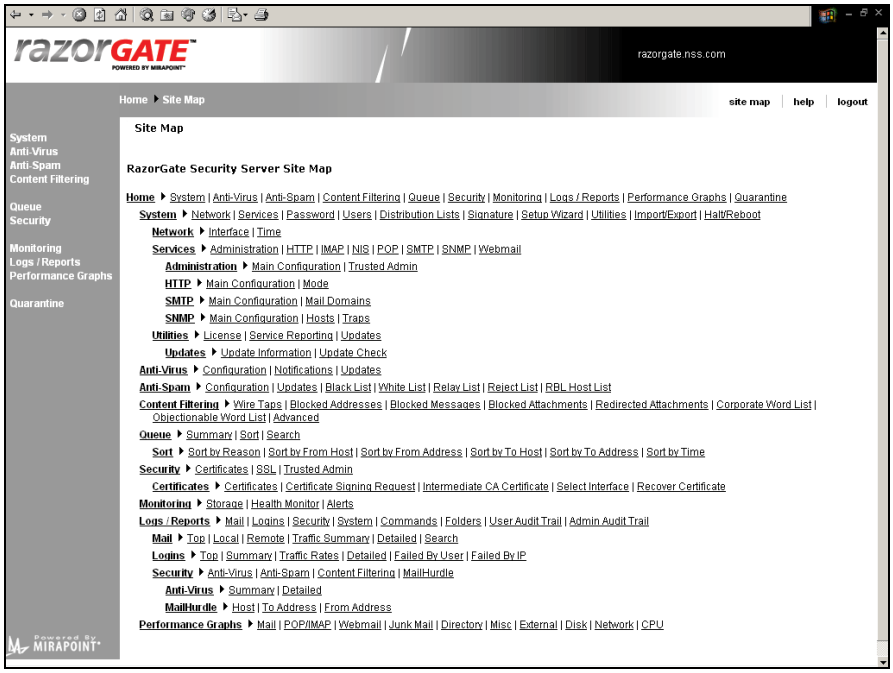


*Figure 3 - RazorGate: Site map*

Note that in addition to filtering mail, the RazorGate system can act as a proxy gateway to existing message stores. The proxy service offers administrators a way to offload SSL processing to the RazorGate from the normal mail server for POP, IMAP and Web-based retrieval of e-mail messages.

In addition, for multi-server environments, the proxy allows administrators to set a single point of entry for e-mail retrieval for the entire company if required. This allows all end-users to have the same configuration for mail settings in their desktop clients, even when multiple different mail servers are used to store their e-mail.

Another advantage of using the HTTP proxy for Web-mail services is the additional protection it provides. Administrators can set limits on directory traversals, header sizes, body sizes and maximum connections in order to prevent common HTTP exploits such as buffer overflows.

Once the system has been installed and configured initially, the administrator has the option to create additional user and administrator accounts on the system. Various roles are available which determine the commands and features which are visible and can be used via the GUI:

- ***Administrator -*** *Can issue all administration commands*
- ***Domain Administrator -*** *Can issue a subset of administration commands*
- ***Helpdesk Administrator -*** *Can issue a subset of administrator commands for handling day-to-day user requests, such as changing passwords and adding members to distribution lists*
- ***Backup Operator -*** *Can issue all commands necessary to perform system backups. This is a read-only role.*
- ***User -*** *Can issue commands relating only to his own account*

Administrators can be restricted to certain IP addresses when accessing the GUI, based on the **Trusted IP** list which is configurable via the GUI.

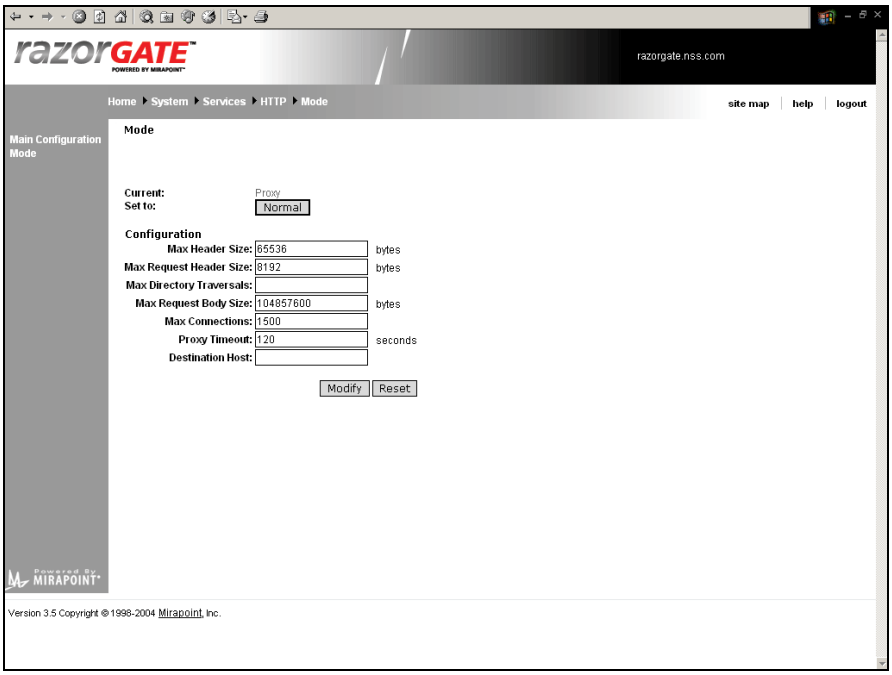


*Figure 4 - RazorGate: Configuring the HTTP proxy*

Frustratingly, only the *Administrator* role can actually be assigned via the GUI, however - all the rest must be configured from the command line. Whilst it is nice to have the CLI available to perform more complex or esoteric administrative functions, we found in general that there were too many more simple and “everyday” operations (like this one) which were not available via the GUI. Mirapoint is working to address this shortcoming in a future release.

## MailHurdle

MailHurdle is one of many Anti Spam techniques used by RazorGate, but one which is so radically different to the others mentioned later in this report that it deserves a section of its own.

MailHurdle is an enhancement to Mirapoint's Full-Spectrum technology which is designed to block spam messages at the SMTP protocol layer before any filtering and analysis techniques are employed.

This offloads network, mail server and gateway processing resources significantly and dramatically off-loads a majority of spam traffic before it enters an organisation's network.

MailHurdle works by only accepting email from RFC-compliant SMTP servers or gateways exhibiting standard SMTP behaviour. MailHurdle does this by challenging the sending server to ensure its RFC-compliance before accepting a connection.

In testing, Mirapoint has seen this technology drop 80 per cent of spam traffic at the SMTP layer, and we noted even higher figures in our lab environment. The total number of messages processed over time dramatically decreases as MailHurdle temporarily fails unknown triplets. After just a few days of using MailHurdle, only 20 per cent of the messages that get through MailHurdle are processed by the rest of RazorGate's Anti-Spam features. When used together with the rest of the Full-Spectrum capabilities mentioned in this report, a total catch rate in the 98 per cent range is claimed by Mirapoint with virtually no additional false positives.

When an email is attempting delivery to a RazorGate system with the MailHurdle feature enabled, the SMTP service looks at three pieces of information, called a *triplet*:

1. *The IP address/domain of the host attempting the delivery*
2. *The envelope sender address*
3. *The envelope recipient address*

In a standard SMTP session, the triplet would be determined as follows:

```
HELO senderdomain.com
250 Hello senderdomain.com #1
MAIL FROM: <sender@senderdomain.com>
250 2.1.0 Sender ok #2
RCPT TO: <recipient@recipientdomain.com>
250 2.1.5 Recipient ok #3
DATA
354 Enter data
```

The SMTP service then determines whether or not that triplet has been seen before. If it has not, a **"Please try again later"** message is returned to the sender. At this point, the message enters an *Initial-Deny* state and a timer is started. By default, the initial timeout value is five minutes and any messages received from that triplet within five minutes are rejected (this prevents spammers from simply blasting the same message over and over again). Most spam mailers will not properly retry addresses that have received such an error code, whereas legitimate mail agents will always retry delivery.

When the five minutes has expired the triplet enters an *Initial-Active* state. If MailHurdle sees the triplet again while it is in the *Initial-Active* state (four hours by default), it accepts the message for delivery and a new *Active* timer starts. Messages from an *Active* triplet are automatically passed through the Mirapoint system, and a triplet stays in an *Active* state for 36 days by default.

During the first *Initial-Active* state, if a triplet is not received by the Mirapoint system within 24 hours, it will be "forgotten" and the MailHurdle process will start over again the next time that triplet is received.

In simplistic terms, therefore, the first time MailHurdle sees a sender it rejects the message (unless that sender is white listed). If the sender is a spammer, it is unlikely that he will retry, and thus the spam is successfully rejected without even having to be accepted and analysed. If, however, the sender is a genuine one, the standard SMTP protocol will ensure that he retries, and the second time the MailHurdle recognises the sender and allows it through for further processing.

Typical spammer tricks such as changing IP addresses to evade black lists will not work, since every mail, and it's delay, is tied to the IP address of the sending relay. If the IP address changes, it effectively “resets” the timer on the delay, even if the envelope sender and recipient addresses stay exactly the same.

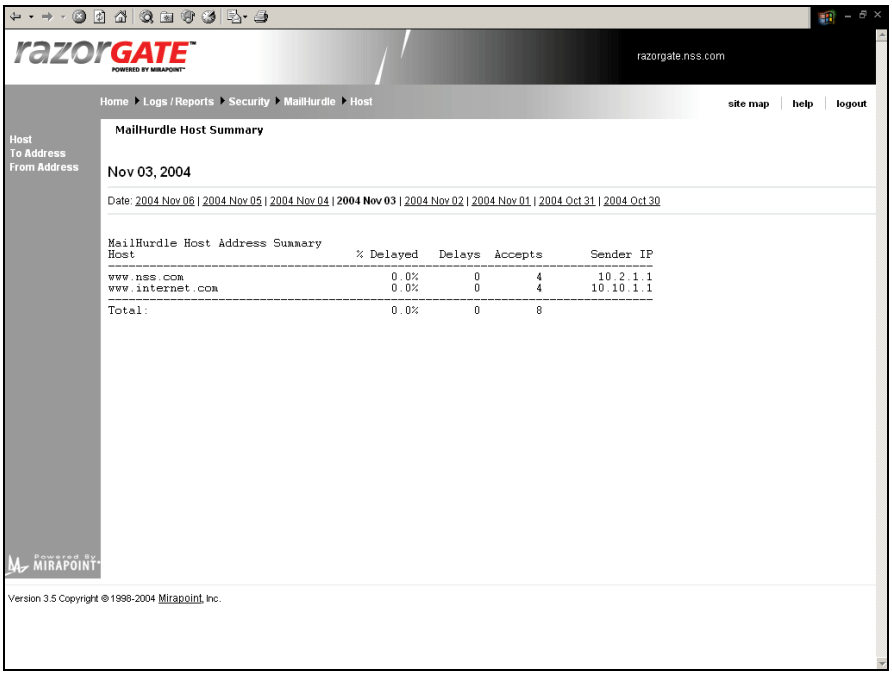


*Figure 5 - RazorGate: MailHurdle reports*

An added benefit of MailHurdle is that it also blocks many viruses that propagate via e-mail using embedded SMTP programs. Many of these viruses often execute directly on a client PC, like in an Outlook client environment, and harvest local e-mail addresses to use as targets for multiplying. Since viruses and their attachments are typically large, bandwidth and processing savings are significant versus the standard method of accepting delivery and performing local virus scans.

The downside to this system, of course, is that there is an inherent delay involved for all genuine mail, since **ALL** new senders are rejected the first time they are seen. This can be overcome by the use of White Lists to ensure that certain domains and addresses are always accepted, though all configuration changes must be made using the CLI - the GUI is restricted to enabling and disabling MailHurdle only.

During testing, this feature was the cause of some frustration when we were waiting for messages from “new” senders to make it through the system to test other features.

This same frustration could occur in the real world if someone is waiting for a specific piece of time-sensitive mail from a new source. But overall, we found the MailHurdle system worked extremely well and it should certainly block a very large proportion of spam today.

MailHurdle can also cause rejection of certain non-spam messages from genuine bulk mailers which are not entirely RFC-compliant (probably for performance reasons) - this can be avoided by using custom White Lists, or by setting a single parameter which white-lists a group of known genuine mail servers that will fall foul of MailHurdle. This list is maintained by Mirapoint and will be applied automatically along with the normal spam updates if the auto-update feature is enabled.

Of course, there is potentially a fairly simple way around MailHurdle as a spam blocking technique, and that is for all spammers to ensure that their bulk mailing systems adhere to the SMTP RFC and retry after a “safe” amount of time whenever they are rejected. Whether this is ever likely to happen is open to debate given the relatively high “cost” to the spammer in terms of additional resources required and time delays waiting to retry - bear in mind that most spammers will be using open relays illegally and therefore their main aim is to get as much mail through there in as short a space of time as possible before they are discovered and shut down.

MailHurdle is probably the feature that currently provides the greatest chance of filtering spam before it even hits the RazorGate system itself, let alone the internal network. Given its potential benefits on resources as a result of rejecting spam without examination **AND** rejecting viruses before they hit the AV scan engine it is well worth deploying.

## Anti Virus

Anti Virus scanning is performed using the Sophos AV engine. We were not asked to verify the effectiveness or performance of the AV scanning, but enough has been written elsewhere about the abilities of the Sophos product - suffice to say that it detected and deleted every virus that we threw at it during the testing period, even when buried within multiple ZIPped attachments. What is important from the Mirapoint angle, however, is how effectively the engine has been integrated into the RazorGate appliance.

Configuration is via two screens in the GUI, the first option of which is to enable/disable the AV protection. Whilst it may seem strange to disable such a vital feature, it should be remembered that each RazorGate security component is a separately licensed product, and some customers may already have a server-based AV solution deployed, and wish to use RazorGate purely as an Anti Spam solution. In this respect, RazorGate is very flexible, since it is possible to enable or disable each of the primary security modules - *Anti Virus, Anti Spam* and *Content Filtering* - individually.

In addition to the above, the first configuration screen provides the ability to specify how infected messages are handled:

- ***Auto Clean (Delete) -*** *Attempt to remove the virus first. If this is unsuccessful, delete the infected attachment*
- ***Auto Clean (Ignore) -*** *Attempt to remove the virus first. If this is unsuccessful, ignore the virus and continue processing the message normally*

- ***Delete -*** *Delete the infected attachment*
- ***Ignore -*** *Ignore the virus and continue processing the message normally*

Virus scanning software distinguishes between two major types of viruses: **cleanable** and **non-cleanable**. A cleanable virus is one that may be removed from a message, document, or program without damaging the message, document, or program. Examples of this type are the macro viruses written in *Microsoft Word* or *Excel* macro language. Some old DOS viruses also are considered cleanable, as are those classified by Sophos as W32/Magistr-A.

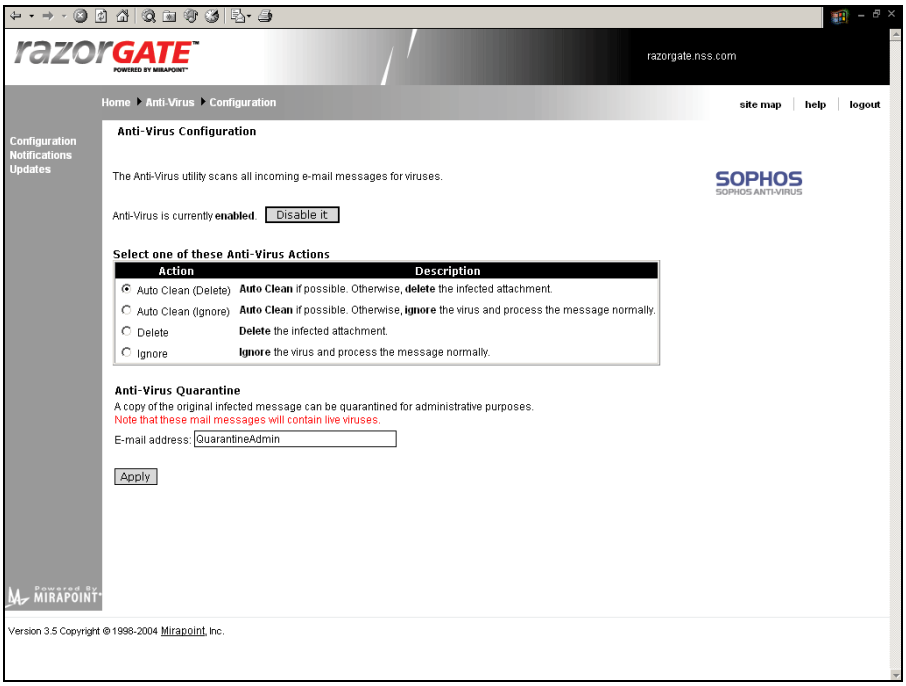


*Figure 6 - RazorGate: Configuring AV scanning*

If a virus is not one of the above, it is considered non-cleanable. In this case, the only way that the message can be made safe is to remove the virus, whether it is the entire attachment or the message body itself. The virus scanner uses pattern files that classify viruses as cleanable or non-cleanable. The system tries to automatically disinfect cleanable viruses if the administrator selects one of the **Auto Clean** options.

After investing in a product such as RazorGate, not many administrators will want to place their faith in the end-users' desktop anti-virus packages and allow infected e-mails to continue on their way into the corporate network, especially since the Sophos engine is not renowned for producing false positive alerts. Thus, the *Auto Clean (Delete)* and *Delete* options are likely to be the most popular.

A copy of each infected e-mail - complete with the live virus - can also be sent to the Quarantine mailbox for further study by the administrator.

The second configuration screen contains notification options. The administrator can choose to send alerts to either, or both, the sender of the infected e-mail and/or the recipient.

It is also possible to specify the text to be included in the notification e-mails, and the in-message notifications when an infected attachment has been deleted. The **From** line, the **Subject** line, and the **Message** text can all be modified for any of the notification messages.

Whilst notifying the sender seems like a good idea at first glance, too many viruses these days are propagated by worms which populate the e-mail address fields with addresses chosen at random from the e-mail address book on the propagating machine. Thus the recipient of the notification is usually innocent, and the notification just becomes another form of spam. In-message notifications are probably enough, so that the recipient knows there was an infected attachment and can take the appropriate action if the sender is someone he knows.

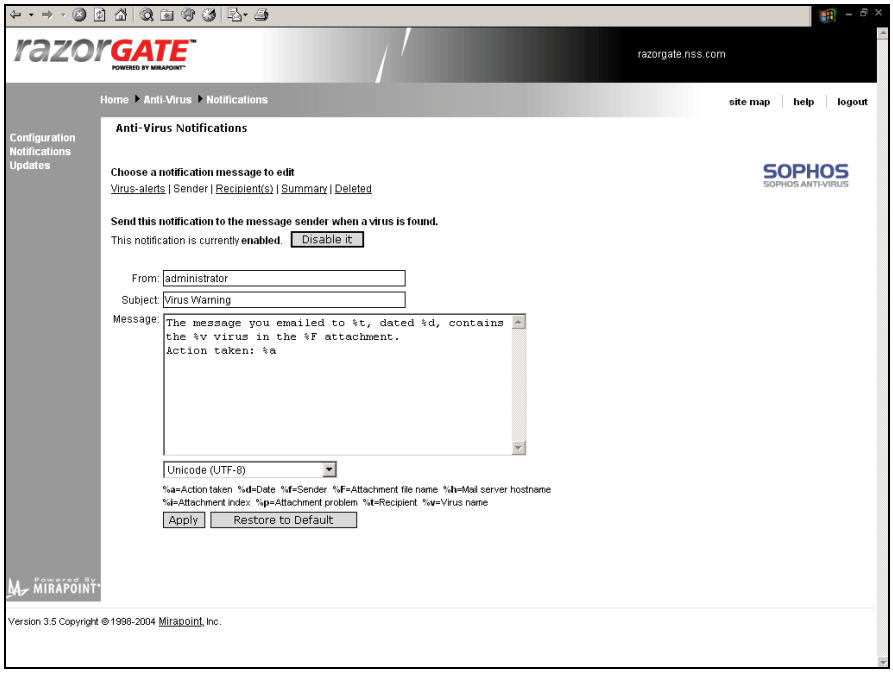


*Figure 7 - RazorGate: AV notification options*

Anti Virus signatures are updated regularly, and updates can be downloaded manually, or scheduled for regular automatic updates on an hourly, daily, weekly or monthly basis.

# Anti Spam

According to Mirapoint, spam control (not Anti Virus) is one of the main reasons most companies purchase the RazorGate. It is testimony to the nuisance that spam has become that Anti Spam is often seen by many as equal in importance to Anti Virus. However, unlike with AV, it is notoriously difficult to write “signatures” for spam - it is too easy for spammers to vary the content of spam messages on an e-mail by e-mail basis if required.

Thus, Mirapoint takes a multi-layered approach to combating spam, with a number of complementary features including:

- *Reverse DNS checks*
- *Sender checks*
- *Recipient checks*

- *Anti-directory harvesting protection*
- *Closed relay protection*
- *Flexible RBL handling*
  - *Reject directly, or tag and process using filters*
  - *Multiple RBL server support*
- *UCE block lists*
  - *SMTP layer protection*
  - *IP or domain name-based protection*
- *Spam analysis engine*
  - *Heuristic and lexical rule checks*
  - *Over 200 rules*
  - *Automatic rule update*
  - *Header and subject tagging*
- *Domain black/white lists*
- *User-level black/white lists (with Mirapoint mail systems)*
- *Personal junk mail folder for quarantine (with Mirapoint mail systems)*
- *Full integration with RazorGate Content Filtering module*
  - *Create customised filters for spam handling based on UCE scores*
  - *Configurable spam action based on multiple thresholds*
- *Adjustable anti-spam threshold*
- *Vipul's Razor support - continuous real-time updates*
- *MailHurdle (see previous section)*
- *Spam reporting features*
  - *Real time logging/reporting/graphs*
  - *Daily security reports*
- *End user spam reporting - Built-in reporting mechanism of missed spam to update rule sets (with Mirapoint mail systems)*

Some of these features (the first five in the list) are built-in to the Messaging Operating System (MOS) at quite a low-level, and thus offer minimal configuration via the GUI. Some of them also depend on the use of Mirapoint (or compatible) systems as the main mail servers, and cannot be effected using RazorGate alone (i.e. user level black/white lists, personal junk mail folder for quarantine and end-user spam reporting). However, the new *Personal Junk-Mail Manager feature* (JMM) is specifically designed to be deployed on the RazorGate without any additional equipment being required. This was released recently by Mirapoint, but not in time to be tested in this report.

Most of the features are configurable at least to some extent via the GUI (with more advanced options configurable via the CLI) and will work with any mail server and client.

Anti Spam scanning uses heuristics and lexical analysis to compare all incoming e-mail messages to a set of rules based on common known factors of junk mail (i.e. looking for all capitals in a subject line, over-use of exclamation marks, repeated use of the word "sex" or "viagra", and so on). Unfortunately, it is not possible to view or tune these rules directly, although reporting spam back to Mirapoint will help to refine those rules, and an automatic update procedure (as with AV signatures) keeps the Anti Spam rules up to date.

For each rule that is matched, a point is added to the message's UCE (unsolicited commercial e-mail) score. The more rules the message matches, the higher the rating it receives, and by default, any message scoring over 50 is considered junk mail and an **X-Junkmail** header is inserted. This header can then be used as a search parameter in a Content Filter on a domain-wide or per-user basis. In addition, an administrator-defined *warning flag* can be inserted at the beginning of the subject line for an immediate visual indication of which messages are considered spam, and also for those situations where the user's mail client is unable to filter messages based on mail headers.

The default UCE threshold is considered to be optimal by Mirapoint, though it is configurable (and we might be tempted to set it slightly lower). Adjusting the *Threshold* to below 50 causes more messages to trigger the *Junk Mail* filter because fewer rule matches are needed. However, this can also mean that more false positives - messages incorrectly identified as spam - will be detected, risking the delay of legitimate e-mail. Adjusting the *Threshold* to above 50 causes fewer messages to trigger the *Junk Mail* filter because more rule matches are needed. However, in this case, the administrator runs the risk of more false negatives - missed spam - being delivered to user mailboxes.

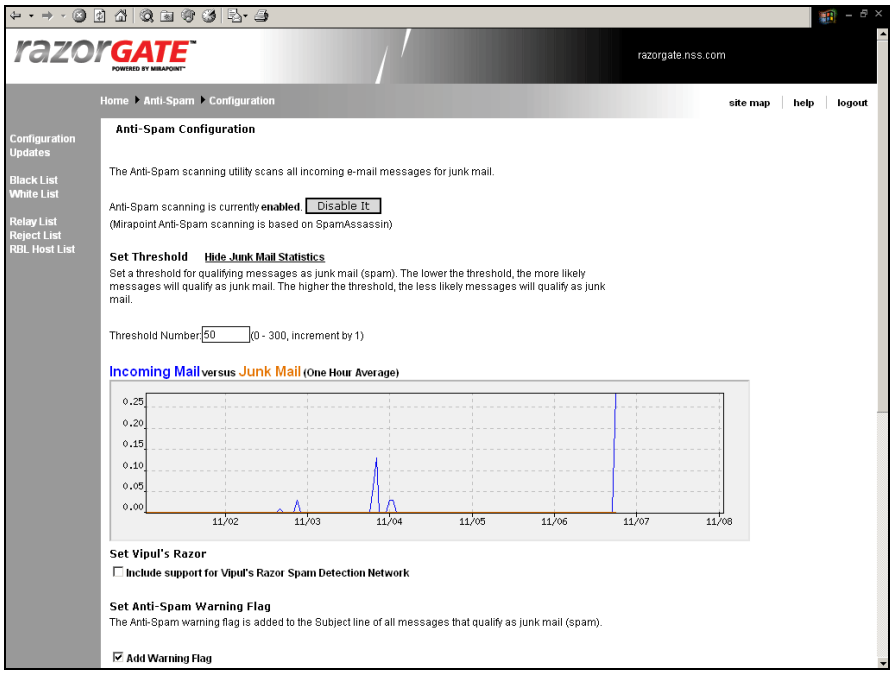


*Figure 8 - RazorGate: Anti Spam Configuration*

It is not always easy to monitor the rate of false negatives once the product is installed, but for those using a Mirapoint mail system the Web-mail facility provides the means to designate individual e-mails as spam. When the user selects this option, he is given the opportunity to add the message sender to a user-level black list, move the message to a junk mail folder, and report the message content directly to Mirapoint support (where it will be used to refine the anti spam rule set).

In addition to the **X-Junkmail** header, which is only added to a message when the UCE threshold is exceeded, an **X-Junkmail-Status** header is added to **every** e-mail scanned.

This contains the UCE score (even when below the designated threshold), and can thus be used by end-users to create their own local junk mail filters at the desktop based on this score. Obviously they are not able to override the system default in order to release e-mails with a score *higher* than the administrator-defined threshold.

As we mentioned at the start of this section, lexical analysis is not the only technique used to identify spam. Indeed, despite advances in this area, this is still probably the weakest technique of those employed, failing to recognise the majority of the spam messages we sent through our system during the test when used alone.

What makes RazorGate particularly effective, however, is the combination of techniques used. The simplest, but often the most effective (despite the risk of “false positives” preventing passage of e-mail from entire domains) is the *Black List*. This is a list of individual source e-mail addresses or domains which the administrator considers to be producers of spam, and from whom all mail should be blocked.

The *White List* overrides the black list, its entries being used to specify those addresses or domains which are considered “good”, and from which no mail should be blocked under any circumstances. The White List can be used, for example, to specify source domains or addresses for important newsletters, which can often otherwise fall foul of Anti Spam scanning systems.

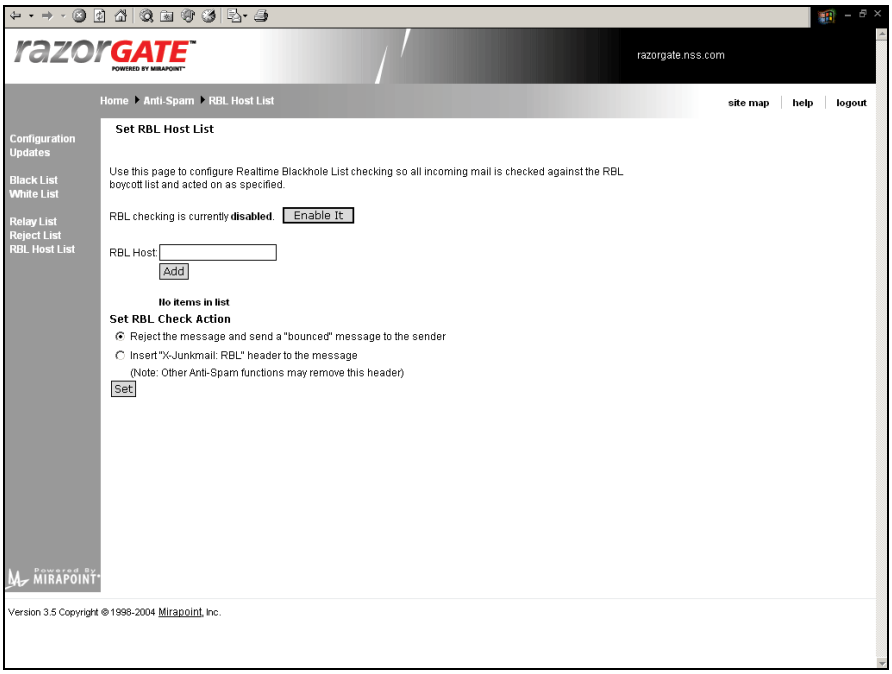


*Figure 9 - RazorGate: Configuring RBL Host List*

Where Mirapoint mail servers are used throughout the system it is possible to employ user-level Black and White Lists which will interact correctly with (i.e. override where applicable) the system-level lists.

Thanks to Mirapoint's history as an e-mail storage server appliance, implementing mail folders is inherent to all Mirapoint products, meaning that per-user junk mail folders are also feasible. By default every RazorGate product has 20 mailboxes, that have POP, IMAP and Web-mail capability.

If desired, the system administrator can purchase a license for additional mailboxes, to allow for per-user storage of junk mail on the RazorGate appliance.

The *RBL Host List* provides the means for RazorGate to query external RBL (Real-Time Blackhole List) servers which will identify source addresses and domains which are known for sending spam. The RazorGate system can either bounce the message immediately, or tag it with a header. The effect of setting RBL Host checking depends on the Anti Spam settings in effect. If Anti Spam scanning is enabled, RBL checking is used to calculate the UCE score, and an appropriate **X-Junkmail** header is added based on that score and any White List or Black List settings. If anti spam scanning is unlicensed or not enabled, messages are categorised as junk mail based on RBL checking alone, and the **X- Junkmail: RBL** header is added.

Note that the Anti Spam system does not block spam mail by default - it is merely used to identify and tag it with mail headers or flagged subject lines. In many organisations, the messages may then be allowed on their way to the end user mail systems for mail server filtering or filtering entirely at the user level (i.e. via Outlook Junk Mail filters, or similar desktop tools). The types of headers added by RazorGate give the end-user fairly precise control over how he processes suspected spam messages, and makes this a very flexible system.

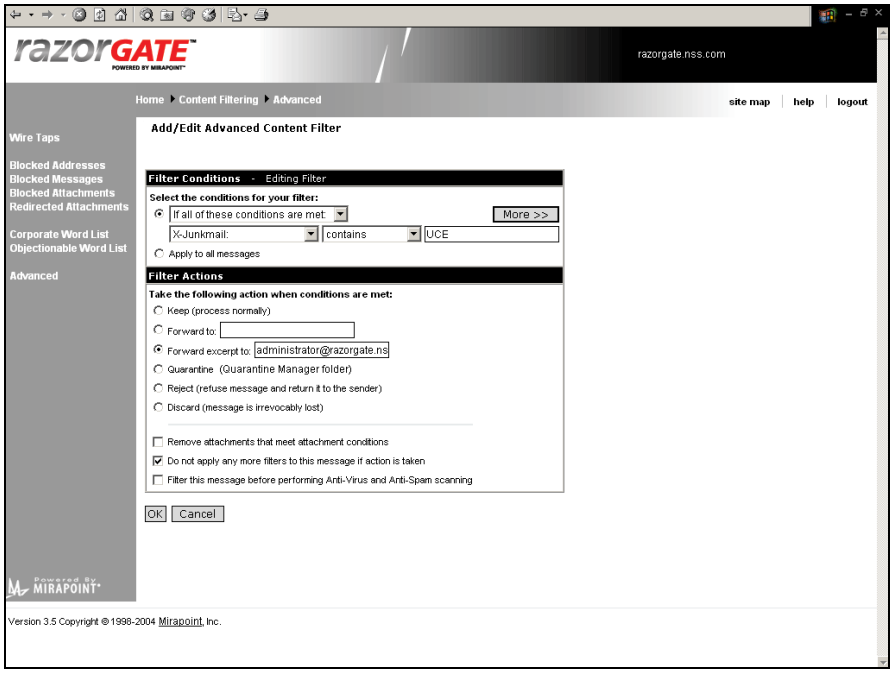


*Figure 10 - RazorGate: Blocking tagged spam messages using Content Filters*

If, however, the administrator wishes to block spam within the RazorGate appliance, the *Content Filtering* module can be used to filter and process messages based on the headers added by the Anti Spam system (see the *Content Filtering* Section).

Another technique used to improve spam detection is support for *Vipul's Razor* (now commonly known as SpamNet).

Vipul's Razor is a distributed, collaborative, spam detection and filtering network which establishes a distributed and constantly updating catalogue of spam in propagation. This catalogue is used by clients to filter out known spam.

Upon receiving a spam e-mail, a *Reporting Agent* (run by an end-user or a troll box) calculates and submits a 20-character unique identification of the spam (a SHA Digest) to its closest *Catalogue Server*. The Catalogue Server echoes this signature to other trusted servers after storing it in its database. Prior to manual processing or transport-level reception, *Filtering Agents* (end-users and MTAs) check their incoming mail against a Catalogue Server and filter out or deny transport in case of a signature match. Catalogued spam, once identified and reported by a Reporting Agent, can be blocked out by the rest of the Filtering Agents on the network.

Other important parameters which can be used to accept or reject mail at the RazorGate are the *Relay List* and *Reject List*. The *Relay List* specifies IP networks or DNS domains for which the SMTP service is to accept messages for relay to remote hosts, thus bypassing the anti-relaying (closed relay) capability of RazorGate for specific hosts.

This is useful for those companies with multiple servers servicing multiple mail domains, all of which pass outbound mail through RazorGate for scanning. The *Reject List* specifies domains and IP addresses from which messages will be rejected completely at the SMTP prompt (i.e. RazorGate will not even accept an SMTP connection from these sources).

When using fully-featured external LDAP servers (rather than the LDAP services built in to the appliance), the RazorGate system also provides *class-of-service* (COS) controls that can define which services are offered to individual users or domains.

The different COS settings can be stored within the RazorGate system or LDAP server, and these controls define which messages are scanned for spam, as well as which end-users have access to the Junk Mail filter and personal White and Black List features. COS can also control which administrators have access to domain-level White and Black Lists. As with other "advanced" areas of the system, COS is outside the capabilities of the GUI, and is administered entirely via the command line interface.

Note that enabling any feature on the RazorGate system can adversely affect performance, especially features such as RBL and Vipul's Razor, both of which can be affected by delays in the external network.

Note also that, by default, Anti Spam scanning is performed on inbound mail only. Whilst this is probably a sensible option for most deployments as a trade-off between security/convenience and performance, scanning for spam on outbound mail is also something that would be useful to many companies. For that reason, we would like to see the option added to the main Anti Spam configuration page in the GUI, instead of being buried deep in the manual and available via an obscure command line option only.

New *Rapid Anti-Spam* technology adds an additional layer of protection for users. Based on the RPD technology licensed from Commtouch, this uses information collected on a global basis from network probes to identify spam and virus outbreaks in real-time.

Since the intelligence behind Rapid Anti-Spam is based on real-time outbreak information rather than analysis of individual message content, the approach should deliver virtually no false-positives, work with any type of message content and language, and effectively distinguish between legitimate bulk mailing and spam. The real-time probe approach also means that no ongoing maintenance or updates are required.

Unfortunately, Rapid Anti-Spam was not available in time for us to test in the lab.

## Content Filtering

The third major component of the RazorGate system is *Content Filtering*, allowing creation and enforcement of corporate policies based on content of messages. Configurable at the domain level or by the end-user, RazorGate filtering allows actions to be taken based on content embedded in the header, body, or attachment of a message.

A number of useful pre-defined filters allow rapid configuration of simple security policies - blocking all suspect attachment types or quarantining messages containing objectionable words or phrases, for example - whilst more advanced customer-specific filters can be created via the GUI and/or the command line interface. All filters can be applied to primary domains only, to inbound mail only, outbound mail only, or to all mail processed by RazorGate.

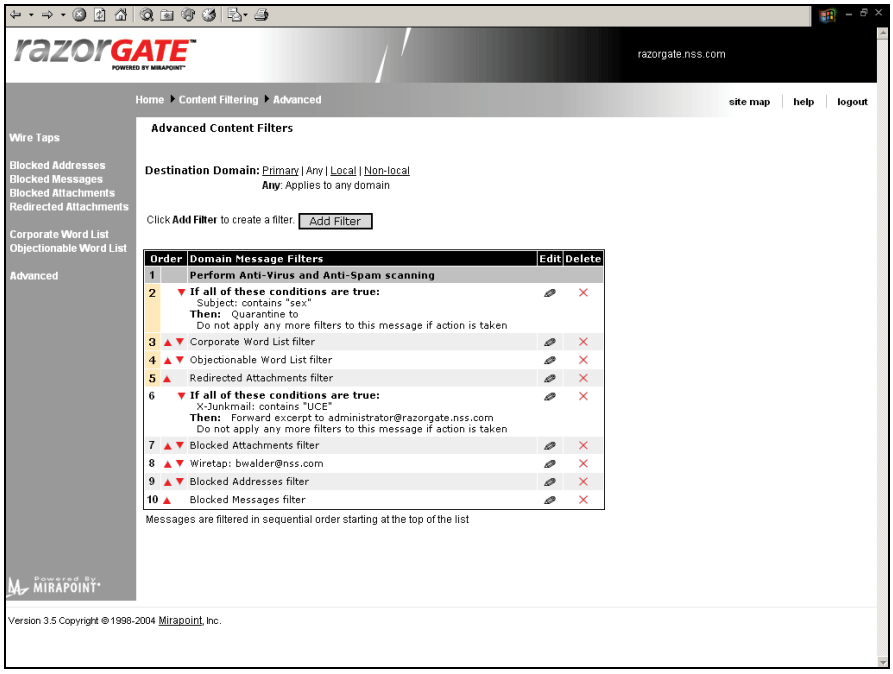


*Figure 11 - RazorGate: Viewing active Content Filters*

These policy-enforcement tools help companies manage employee e-mail usage, prevent leaking of sensitive information, address regulatory compliance requirements, and protect against harassment and other types of e-mail abuse. However, if any of these areas are critical, it should be noted that the RazorGate system does have some significant restrictions.

For example, although it can scan simple attachments such as text documents, Word documents or Excel spreadsheets, it is not capable of scanning attachments that have been compressed as ZIP or TAR files.

But for many organisations, the RazorGate features will be adequate for their needs. One nice feature which will put fear into the hearts of any misbehaving employee is the *Wire Tap*. As its name suggests, this provides the means to copy all messages (inbound and outbound) for a specific user to a separate mailbox for review. Whilst this may not sound like rocket science - after all, most standard mail servers have the ability to copy specific mail streams in a similar way - RazorGate's Wire Tap has one very important advantage.

Traditional duplication mechanisms treat the target mailbox just like any other, and thus if there should be a problem with that mailbox (i.e. if it is on another server which suddenly fails, or the mailbox becomes full) then the copied e-mail could be "bounced" back to its sender, which could well be the actual user under investigation. Obviously this gives the game away completely, alerting the user that he is under investigation. RazorGate's Wire Tap eliminates this problem by re-writing the envelope as it duplicates each message and also allowing the administrator to specify where any bounce messages should be sent.

The *Blocked Address* filter provides a simple means to block all e-mail from a particular domain or individual e-mail address (messages can be rejected or silently discarded). Thus it is possible to quickly prevent a disgruntled ex-employee or a rival organisation from communicating with your employees. Unfortunately, this is a one-way only filter, so it does not prevent outbound communication to banned addresses. Although it would be a simple matter to create a custom Advanced Content Filter to block mail in both directions, it seems a trivial oversight for a built-in filter that could otherwise be used frequently.

The *Blocked Messages* and *Corporate Word List* filters are very similar in operation in that they allow the administrator to create lists of words or phrases which, when found within a message or attachment, will trigger an action. The action is the only thing which differs between these two filters - the *Blocked Message* filter offers a reject or silent discard option, whilst the *Corporate Word List* filter allows the suspect message to be forwarded to another mailbox, quarantined, rejected (bounced back to sender) or silently discarded.

Both of these filters can obviously be used to go some way towards enforcing certain regulatory requirements - such as financial institutions rejecting any outbound messages containing the word "*guarantee*" - or protecting confidential information - such as rejecting messages containing words such as "*confidential*", "*proprietary*" or "*NDA*". However, the slightly different responses offered by each filter would allow the administrator to apply different policies for different types of words. In general, we feel that the Corporate Word List filter would be more widely used since it offers the option to quarantine messages for further review - after all, a filter such as this would tend to be prone to false positives.

The *Objectionable Word List* filter is almost identical to the Corporate Word List, except the use of a separate list provides the opportunity to treat potentially offensive e-mails differently to those which may pose a threat to corporate confidentiality.

This filter, too, provides options to silently discard messages, reject then, forward them to another mailbox, or Quarantine them.

The *Quarantine* mailbox is a separate mailbox which is available to a specific user only - the *Quarantine Administrator* - and which has its own Web-mail interface for processing.

This is like the standard Web-mail interface offered by the RazorGate and other Mirapoint mail systems for normal users, but with two additional buttons added to the interface - *Delete* and *Approve*. *Delete* simply discards the message and removes it from the Quarantine inbox, whilst *Approve* forwards the message to its intended recipient. In both cases, neither the sender nor the recipient are made aware that the mail was ever held up in quarantine.

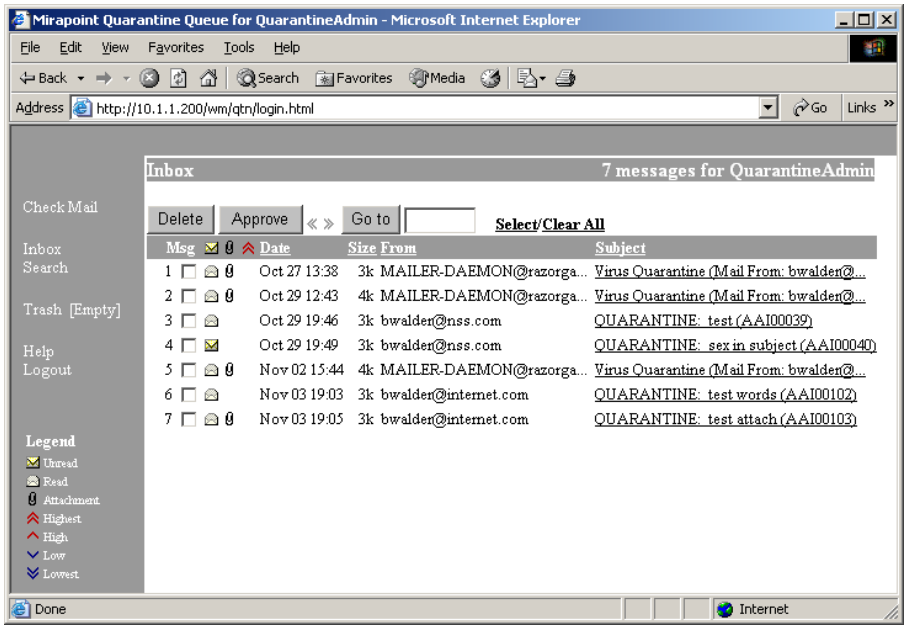


*Figure 12 - RazorGate: Quarantine mailbox*

The Quarantine idea is good - in fact it is essential when doing extensive Content Filtering since the opportunity for false positives is typically high when *Objectionable Word Lists* and *Corporate Word Lists* are in play. However, we do not feel that the Quarantine processing has been implemented well in the current version.

The Web-mail interface is too limited, in that it presents only the *Date, Size, From* address*, attachment* indicator*,* and *Subject*. There is virtually no indication as to ***why*** the message has been quarantined, other than some rather wide-ranging labels that are tagged onto the beginning of the *Subject* line to tell you whether the mail has been quarantined by the AV system or the Content Filtering system.

Whilst that is useful as far as it goes, the biggest problem lies in determining why the Content Filtering system has quarantined a message - was it because it violated the *Corporate Word List* filter, or the *Objectionable Word List* filter, or the *Blocked Message* filter, or one of the potentially hundreds of custom *Advanced Content Filters*? And what exactly within the mail message or its attachments triggered the violation? It would be nice to have the suspect word or phrase highlighted within the message on screen somehow.

It would also be very useful to have the recipient of the message shown on screen when scanning the inbox contents, rather than being forced to open the message to glean this non-trivial piece of data. It is often very difficult for the Quarantine administrator to make a sound decision given the overall lack of information presented.

Finally, in the current version, the Quarantine option is only available to filters which have been applied to **all** mail, and not when a filter is restricted to the primary domain, or inbound/outbound mail only. Mirapoint accepts that this is one area of the product which needs some work, and has committed to addressing these points in a forthcoming release.

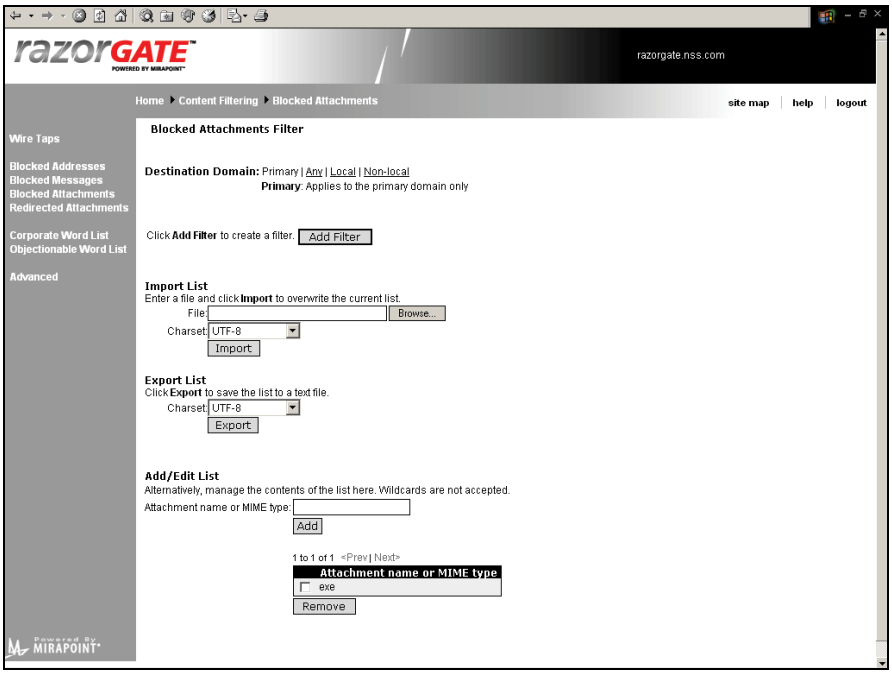


*Figure 13 - RazorGate: Block Attachments filter*

The *Blocked Attachments* and *Redirected Attachments* filters both allows the administrator to specify a list of known harmful attachment types by file extension (EXE, VBS, SCR, PIF, etc.) or by MIME type.

As the names suggest, the *Blocked Attachments* filter provides options to silently discard or reject the message with the suspicious attachment, whilst the *Redirected Attachments* filter provides options to forward the message to another mailbox or Quarantine it. In both cases, it is the entire message that is acted upon, not just the attachment, and no notification is sent to the sending or receiving parties as would happen if the AV system discovered an infected attachment.

Both of these - and *Blocked Attachments* in particular - provide a good “front end” protection mechanism which can eliminate many infected messages almost immediately they are received, and before they are subjected to the more resource-intensive process of AV scanning.

Finally, for those who need to create their own policies and address needs not covered by the pre-defined policy filters, there is the *Advanced Content Filter* option.

Here, the administrator gets to specify one or more conditions (up to 10 via the GUI, and up to 20 via the CLI) to be met when matching against mail messages (multiple conditions can be "*match all*" or "*match any*") and the corresponding actions when matched:

- *Keep (process normally)*
- *Forward to specified mailbox*
- *Forward excerpt to specified mailbox (forwards first 16 lines of the message)*
- *Quarantine*
- *Reject (refuse message and return it to the sender)*
- *Discard (silently drop the message)*
- *Remove attachments that meet attachment conditions*
- *Do not apply any more filters to this message if action is taken*
- *Filter this message before performing Anti-Virus and Anti-Spam scanning*

Each filter can have only a single action assigned to it, but it is possible to create multiple filters for the same condition and give each one a different action, having them processed sequentially in a specified order (filters can be re-ordered on-screen).

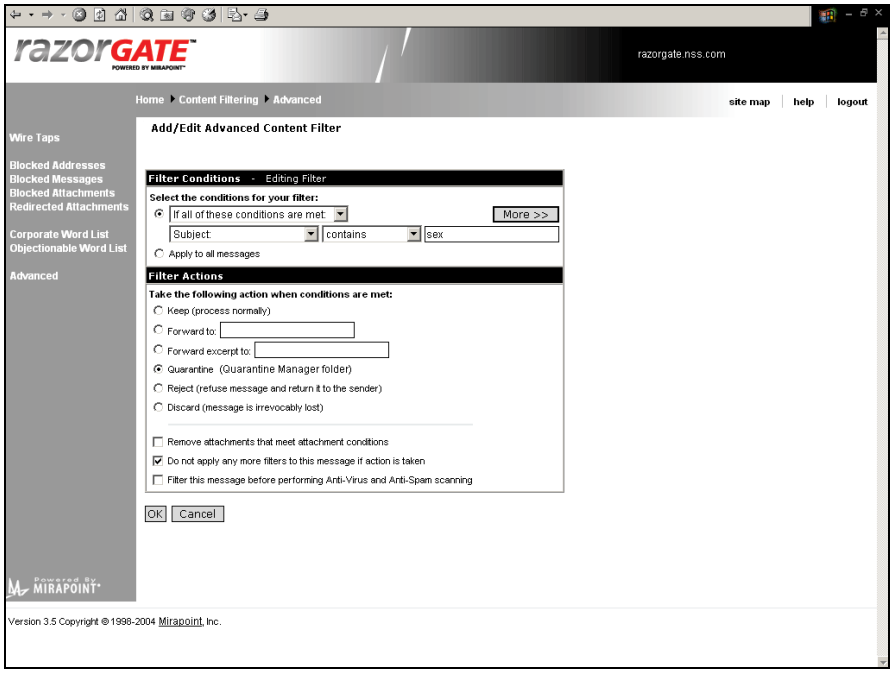


*Figure 14 - RazorGate: Creating Advanced Content Filters*

A wide range of header and message attributes can be matched within each e-mail (and its attachments) including *from address, to/CC address, subject, message body* (including attachments), *junk mail status/UCE score, virus status* (action taken by the AV scanner on infected messages - ignored, deleted, quarantined), *attachment file name, white list/black list status, message size* (bytes), and many more.

More complex filters can be created via the command line, and basic wildcards can be used in certain cases.

Whilst regular expressions are not supported by the current system, this will change in the next release, providing an even more powerful means of filtering based on mail content.

## Additional Security Features

In addition to the main security components discussed already, RazorGate has a number of additional, hidden, and mainly non-configurable features designed to protect the system itself and all associated data.

These features include:

- *Anti-Phishing capability*
- *Directory Harvesting Protection*
- *Password Harvesting Protection*
- *Host-based Intrusion Detection*
- *Denial of Service protection*
- *Anomaly Detection and Prevention*
- *SMTP exception settings*
- *Real-time event logging*

As part of the heuristics and lexical analysis engine, Mirapoint has incorporated a set of rules to determine when a message is likely to be a “phishing” attempt, and automatically tag such messages to be processed as the system administrator has indicated for other types of spam messages. These rules were developed independently of the spam engine using large numbers of phishing emails collected by Mirapoint from its own honeypots and various end-users of the system. These rules look at specific characteristics around phishing messages including (but not limited to) the URL's that are shown to the end-user and the underlying HTML used in the message.

As a self-protection mechanism, RazorGate is designed to automatically detect when anomalous behaviour occurs from single IP address, and act accordingly to mitigate the threat. For example, if an SMTP connection occurs repeatedly from the same IP address, or a login attempt occurs repeatedly and each connection has some anomaly - either a syntax error in a command, repeatedly sending to unknown recipients (directory harvesting), or a bad password is offered multiple times - RazorGate will automatically introduce delays for future connections from that IP address.

RazorGate will also continually increase the amount of delay introduced with each subsequent anomalous connection - thus implementing connection throttling for suspicious connections whilst allowing normal connections to proceed as usual. These events are also logged, both real-time and in the daily logs, for administrative review and alerting.

The use of a closed, hardened operating system as the base for RazorGate should hopefully lessen the chances of successful illegal access. However, nothing is certain in this world, and so RazorGate has implemented a *Host-based Intrusion Detection System* (HIDS) designed to alert the administrator when illegal access is made to closed parts of the operating system.

# Monitoring and Logging

RazorGate contains extensive monitoring and logging capabilities, and benefits greatly from being a tightly integrated hardware and software platform. This is no more evident than in the disk management capabilities.

Whereas some other vendors force the administrator to use third party utilities to monitor, configure and rebuild RAID devices (i.e. the utilities provided by the vendor of the RAID controller) Mirapoint has integrated this functionality right into the management GUI.

The first option in the *Monitoring* menu is labelled *Storage*, and this brings up a graphical display of the six drive bays in the RazorGate appliance, with colour coded pictures of each individual drive showing which ones are included in an array, which are hot spares, which are in use, which are failed, and so on. Usage statistics can be retrieved for individual drives or the entire array. It would be nice to be able to set high watermarks for drive storage to have the system raise alerts should the array fill to a pre-defined level.

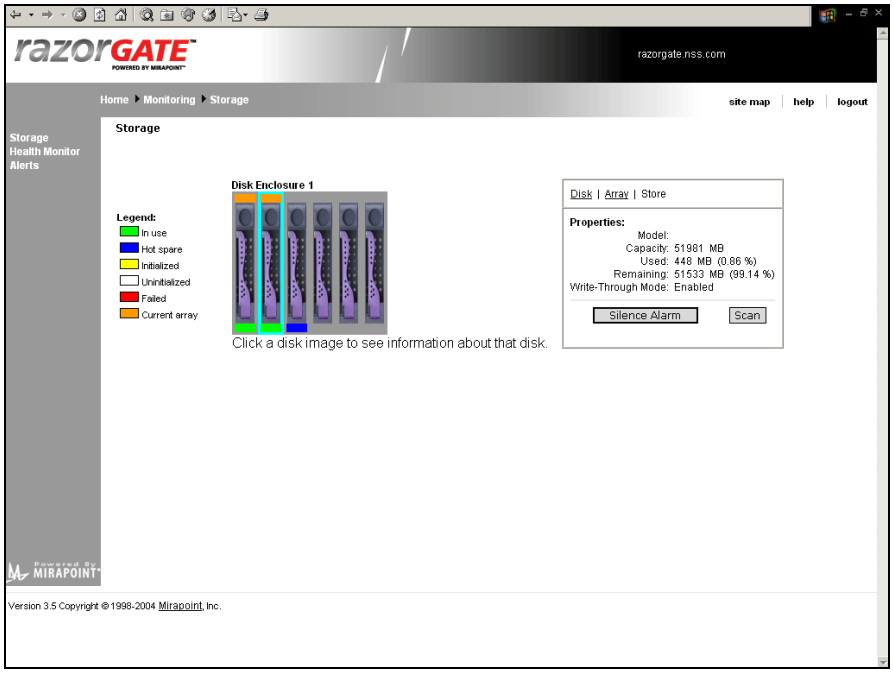


*Figure 15 - RazorGate: Monitoring disk storage*

Should a drive fail for any reason, the administrator is alerted via system alerts (and SNMP traps if configured) and the failed drive can be replaced whilst the system is live with no interruption in service. Customers with a support contract will be shipped a replacement drive immediately with no intervention, since alerts are also sent to Mirapoint Customer Care (if configured). Once the drive has been replaced, the *Storage Monitor* screen provides the ability to scan for the new drive and rebuild the array without having to leave the GUI or resort to using external tools.

The *Health Monitor* provides a summary of critical components within the system and their conditions - such as CPU status, CPU temperature, fan status, ECC errors, power supply status, system temperature, and so on. Should any of the components fail, this is indicated on this screen, and alerts are raised.

The alerts are transmitted by e-mail (and SNMP traps if configured) and also appear on the *Monitor-Alerts* screen.

## Performance Graphs

There are a large number of *Performance Graphs* to provide at-a-glance indication of the load on the system, the level of spam being caught, number of viruses detected, and so on.

The main dashboard display provides three dials showing system load, CPU utilisation and mail queue size. Graphs can be configured to show results for the last day (default), the last hour, or an instantaneous reading, and can also be set to automatically refresh.

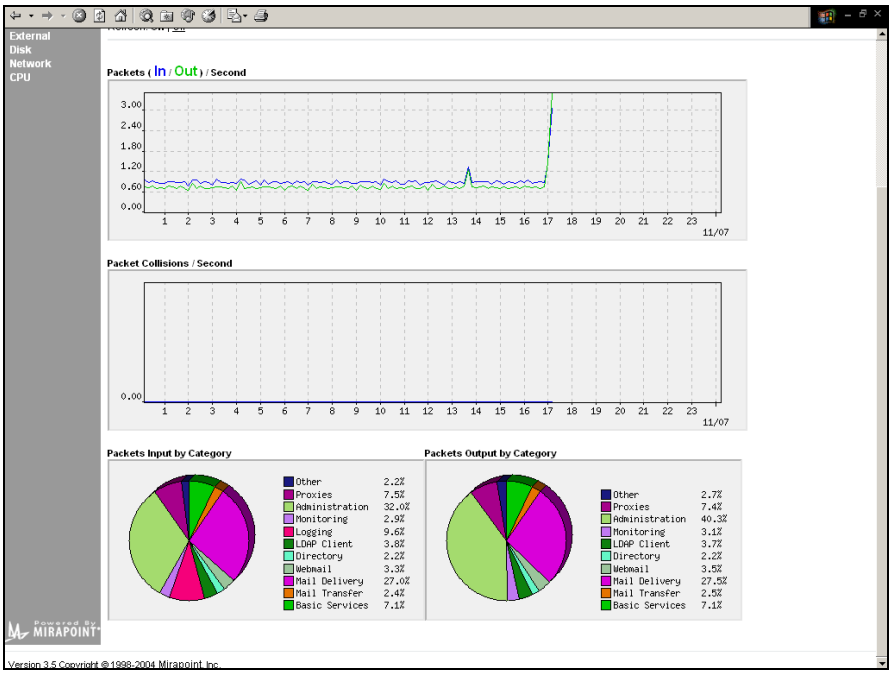


*Figure 16 - RazorGate: Performance graphs*

The Dashboard provides an instant high-level view of which area of the system needs attention, and from there, the remaining Performance Graphs can be used to drill down to the detail.

It would be nice if this were a real “drill down” display - where clicking on one of the dials immediately took you to the more detailed report of the data behind that dial - but that is not the case here. Each report must be called up separately and manually.

Though not directly concerned with the RazorGate device itself, another extremely useful monitoring capability is *External Server Monitoring*. Here, the Mirapoint MOS automatically monitors all external servers which can be considered critical to the operation of the appliance, such as the DNS server, router (default gateway), LDAP server, and NTP server. Graphs showing the real time response time of these servers are available on the Performance Graphs page, allowing the administrator to pinpoint quickly which external server may be causing delivery or response time issues for the RazorGate appliance.

Other useful graphs show resource usage in pie chart form. Here, the individual “wedges” of the pie represent the various processes or protocols. For example, the disk read/write activity chart shows which of the main subsystems are hogging the disk - Anti Virus, Anti Spam, Logging, Webmail, Message Database, and so on.

## Logs & Reports

Behind the graphical summary views are extensive audit trails and log files, and the *Logs/Reports* menu option provides access to those. Logs can be viewed historically or in real-time, and fixed menu options provide access to summary reports or detailed views, and to specific time periods (usually daily).

A search function allows very basic searching for a single text string within the detailed mail logs.

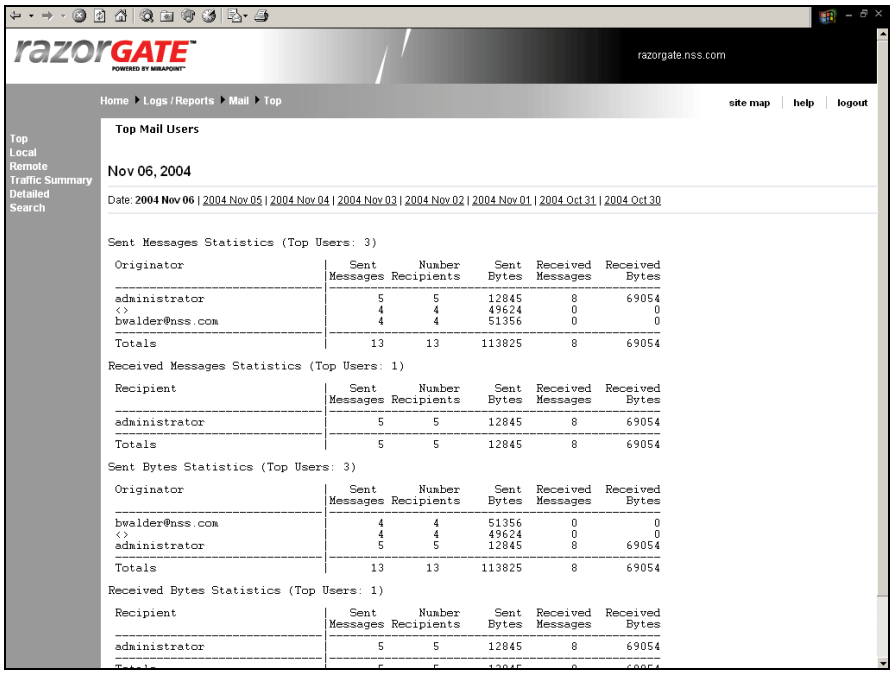


*Figure 17 - RazorGate: Typical report*

Summary log reports are e-mailed to the administrator at regular intervals, so it is not even necessary to access the GUI in order to be kept informed of how the system is performing.

Although the logging and monitoring facilities are very comprehensive, the one thing that is missing from RazorGate at the moment is any form of real reporting capability.

None of the “reports” within the system provide any formatting or other customisation options, and the content, sort order and frequently the start and end times are all fixed. The type of reports available from the system at present are limited to very high level summaries of total number of messages by protocol, scanned by AV, scanned by Anti Spam, and so on.

There is no means to create reports that show a mail summary to/from a particular domain or e-mail address, for example.

Or how about being able to list the total number of messages, by sender and/or recipient, which have triggered each Content Filter, and how many of those messages were subsequently Approved or Deleted?

This type of report would be invaluable in determining over time if a particular filter is being bothersome in raising too many false positives.

## Queue Management

Finally, an essential part of any mail system is the ability to monitor and manage the mail queue - a queue build-up can indicate the outbreak of a new virus or worm, a major spam disbursement, a Denial of Service attack, or a major outage in either the Internet or other network component. But whereas full queue management is expected of any typical mail server, it is a shock to see how many mail security gateways seem to overlook this vital feature. Not so with RazorGate.

As with other areas of the system, the *Performance Graphs* provide vital insights into the operation of the mail queue, but the real strength of RazorGate is the dedicated *Queue* menu option. The informative *Summary* screen includes the longest time a message has been in the queue, total number of entries in the queue, the receiving host with the maximum entries, receiving recipient with the maximum entries, sending host with maximum entries, and sending user with maximum entries, as well as the most common reason for a message being in the queue.

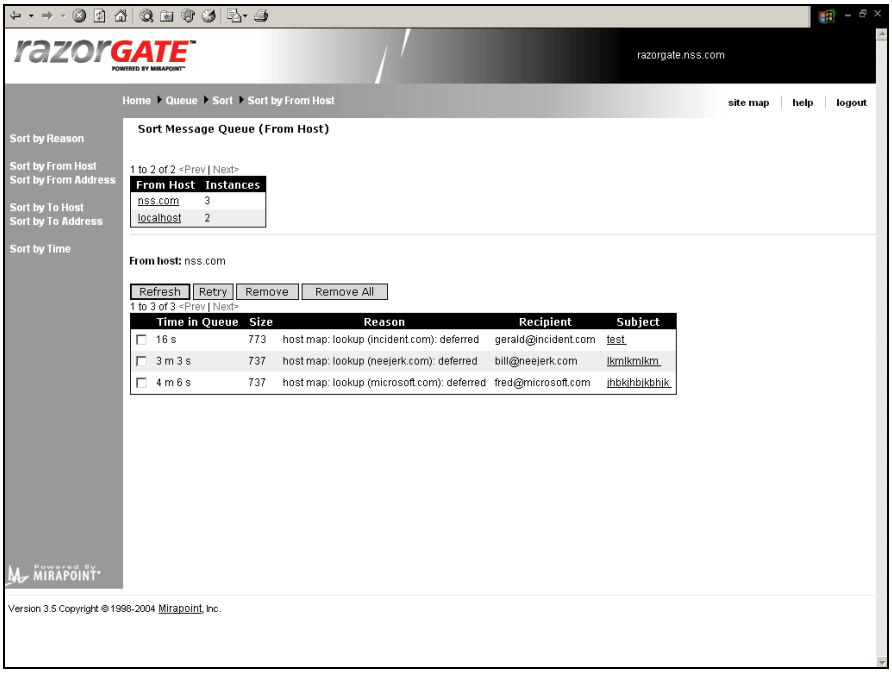


*Figure 18 - RazorGate: Queue Management*

The detailed *Message Queue* screen provides a number of different sorting options, including:

- *Reason for being queued*
- *From host/address*
- *To host/Address*
- *Time in queue*

On selecting one of the sort options, a summary table is displayed at the top of the screen showing total messages in the queue against each group entry - for example, on selecting "*From Host*", a table appears showing each source host in the queue along with the total number of messages from those hosts currently in the queue.

Selecting any one of the group entries then displays the individual queue entries for that group, and provides the option for the administrator to retry delivery or delete from the queue altogether. In addition, clicking on the *Subject* line of any message in the queue brings up detailed information on that message.

Finally, a Search function allows the administrator to search through the queue using a number of different criteria (including *queue ID, minimum time in the queue, reason* and *recipients*) to locate messages that have similar characteristics not covered by the pre-defined menu options. Once the results are presented, the administrator has the same options as with the sort screens mentioned above, allowing him to select messages for retry, removal, or closer inspection.

## Verdict

Despite the potential complexity of the product, RazorGate is extremely straightforward to install and configure. The documentation is outstanding throughout - extremely detailed and well written - and the on-line help, although not context-sensitive, is also extensive and useful.

Some of the RazorGate features (not many) depend on - or work much better with - the use of Mirapoint mail servers in conjunction with RazorGate. Potential purchasers would do well to ensure that **all** of the features in which they are interested will work in the manner intended with whatever mail servers are already in place. Incompatibilities should be few and far between (since everything is standards-based), but certain features - such as end-user spam reporting via Web-mail - will obviously only be available when using Mirapoint systems throughout.

Given that Mirapoint can provide everything from boundary security down to high-performance, high-capacity core mail server appliances, this is a powerful incentive to opt for an all-Mirapoint solution. However, if the thought of a fork-lift upgrade for all your existing mail servers is just too daunting to contemplate, RazorGate will work perfectly well as the boundary security solution for any existing mail system.

As a complement to the AV and Anti Spam capabilities, Content Filtering is adequate - but those who require extensive content filtering capabilities should explore the use of more advanced stand alone products. For example, the RazorGate Content Filtering is basic in its handling of attachments, since it cannot look inside ZIP or TAR files.

We feel the majority of users will buy RazorGate for its Anti Spam and Anti Virus capabilities, and will treat Content Filtering as something of a "bonus". This is not to say that Content Filtering is useless - far from it, especially with the *Advanced Filters* option. But whereas both Anti Virus and Anti Spam could be considered feature rich and even "best of breed", the same cannot be said of Content Filtering - better options exist in the market if Content Filtering is critical to your organisation.

Anti Virus scanning  is based on the Sophos engine, which is proven technology, and which worked extremely well in all our tests. Configuration of the Anti Spam system is more extensive than for AV, but certainly no more difficult. Most of the difficult options are well hidden behind simple check boxes in the GUI, and the more complex and obscure configuration capabilities are restricted to the CLI.

Although the lexical analysis capabilities alone were fairly average in detecting spam, we think that once all options have been configured - especially with *MailHurdle*, which was an excellent feature - the spam catch rate should be very high with RazorGate in most environments.

Although we did not perform any extensive performance benchmarking as part of this test, we did run many tests to ensure that the AV, Anti Spam and Content Filtering options were working as expected. Clearly such computationally intensive processes as Anti Spam and Anti Virus scanning (the latter in particular) do not come without a performance impact, but once the system had been configured intelligently (i.e. with *Content Filtering* removing prohibited attachments so that AV no longer had to scan them) we noted no severe performance penalties (other than when priming the *MailHurdle* system where the initial delays following rejection of first-time e-mails are inevitable).

Where performance is critical, potential purchasers would be wise to perform their own capacity planning tests on their own network, and consider the use of load balancers in order to scale the RazorGate solution accordingly.

All in all, we found the RazorGate system to be very impressive. Feature rich and very capable in all departments, it offers an extensive multi-layered approach to security rather than relying on a single technology (even within the individual subsystems - Anti Spam being the most obvious example).

However, at no point does it give the impression that this is a collection of disparate security offerings that have been bolted together in a haphazard manner. Everything from the hardware, to the software, to the management interface gives the impression that it has been designed from the ground up with both ease of use and security in mind.

This, along with the fact that Mirapoint can offer a range of appliances for all requirements, means it is worthy of consideration in any situation - from small business to service provider - where e-mail security is taken seriously.

## Contact Details

**Company name:** Mirapoint Inc.

**Internet:** www.mirapoint.com

**Address:**
909 Hermosa Court
Sunnyvale
CA 94085
USA

**Tel (Sales):** +1-800-937-8118
**Tel (General):** +1-408-720-3700
**Fax:** +1-408-720-3725

**E-mail:** info@mirapoint.com

**Company name:** Mirapoint Europe Ltd.

**Address:**
8 The Square
Stockley Park
Middx.
UB11 1FW
United Kingdom

**Tel :** +44-(0)20-8610-6044
**Fax:** +44-(0)20-8610-6855

**E-mail:** dl-europe@mirapoint.com