Global Information Assurance Certification Paper

Creating a Secure Email Gateway with Qmail/Qmail-Scanner

Mike Sullivan

GSEC Version 1.2f

02/27/2002

I. Introduction

One of the biggest security concerns facing companies today is the outbreak of malicious email viruses and/or worms. According to a recent report from Computer Economics, the top outbreaks over the last 3 years have had an impact of approximately 42.4 billion dollars. Furthermore, a recent survey by ICSA Labs demonstrates that email is now the biggest vector for the spread of malicious code.

ANALYSIS BY INCIDENT

YEAR	CODE NAME	ECONOMIC IMPACT
2001	Nimda	$635 Million
2001	Sir Cam	$1.15 Billion
2000	Love Bug	$8.75 Billion
1999	Melissa	$1.10 Billion
1999	Explorer	$1.02 Billion

Excerpt from Computer Economic report, "2001 Economic Impact of Malicious Code Attacks"

ANALYSIS BY YEAR

YEAR	ECONOMIC IMPACT
2001	$13.2 Billion
2000	$17.1 Billion
1999	$12.1 Billion

Excerpt from Computer Economic report, "2001 Economic Impact of Malicious Code Attacks"

SOURCE OF INFECTION

SOURCE	1996	1997	1998	1999	2000
Diskette	74%	88%	67%	39%	7%
Email	9%	26%	32%	56%	87%
Download	12%	18%	12%	13%	2%
Other	15%	20%	11%	13%	4%

Excerpt from ICSA Lab report, "6th Annual Computer Virus Prevalence Survey 2000"

A sound email attachment filtering policy, coupled with a good commercial antivirus scanner, would have easily stopped the spread of these viruses. When they are not stopped, the impact can be severe. Of course, there is more to it than just the economic damage; there is also the loss of credibility and confidence between the company and the customer. This paper will present one scalable, flexible solution to one small businesses attempt to take control over its email infrastructure.

II. Case Scenario

This scenario involves a small company (approximately 450 employees) that has 11 offices located around the country. Each office has its own mail server; however, they are not standardized on one mail server package. One location is running Exchange, some are running Sendmail on Sun/Cobalt Qubes, and some are running Qmail on plain Linux servers. This makes the task of standardizing an antivirus/attachment-blocking solution a real nightmare. The commercial antivirus solution for this company is McAfee/NAI's Total Virus Defense (TVD) product. This has a solution (GroupShield) for the Exchange server and a command-line scanner for Linux (uvscan). It does not include any products that work directly with Qmail or Sendmail. Furthermore, Sun/Cobalt does not provide any antivirus support for their sendmail installations on the Qube. What was needed was a solution that would allow centralized antivirus management and updates, as well as application of a uniform attachment blocking solution across all mail servers in the company. This would allow for a central chokepoint in the event of a new outbreak as well as less overall administration in applying policy.

III. Solution

The solution developed in this case involved implementing gateway servers for all email entering and exiting the company's mail system. This would provide a single chokepoint for all email, a single point for antivirus management, and a single point for applying attachment-blocking policy. The final plan was a Linux based solution consisting of Red Hat Linux, the Qmail MTA package, the Qmail-scanner Perl script, and the command-line scanner from McAfee/NAI (uvscan). Because of the company's size, the solution needed to be inexpensive and easy to maintain. This made Red Hat an easy choice for the base operating system. Qmail was chosen as the mail server package because it is a highly secure mail server, and it is also quickly becoming one of the most widely used MTAs on the internet. Finally, Qmail-scanner was selected as the content scanner as it is a highly configurable Perl script that integrates nicely with Qmail and Uvscan. It also allows for scanning of local email and, more importantly, email crossing the server (relayed email).

IV. Implementation

For the sake of redundancy, you may want to configure more than one gateway server to meet your needs. This document will cover setting up just one server, but you can add as many as you need for your particular scenario. The domain name used throughout this document will be sullys.net, and the specific host will be called sans.sullys.net. This domain name was chosen so that it would not reveal any sensitive company information. There is also a download site on www.sullys.net with links to all of the software required to complete this installation, with the exception of Red Hat. The links are provided in Appendix D, however it is best practice to download the software from the official website whenever possible. This ensures that you have the latest version available.

A. Requirements

The following software packages and patches are required to complete this installation:

Red Hat Linux 7.2
Daemontools
UCSPI-TCP
DJBDNS
Qmail 1.03
Qmail-queue Patch
Maildrop 1.3.6+ (reformime)
Perl 5.005_03+
Perl module Time::HiRes
Perl module DB_File
TNEF 1.1.1
Commercial antivirus scanner (Uvscan in this case)

A listing of all download locations is located in Appendix D, and the location of each package is also listed in the relevant installation instructions.

B. Installation

1. Install Red Hat Linux 7.2

The process begins with a clean installation and hardening of Red Hat 7.2. The server used in this case is a PIII/450 with 128 MB RAM and a 4.3 GB IDE hard drive. A default "Server" installation with classic X-server was chosen, with the exceptions noted below:

Custom Partitioning:

/	2.5 GB
/boot	50 MB
/tmp	500 MB
/var	1.0 GB
<swap>	256 MB

Custom Firewall:

Allow SMTP, SSH

For the customize firewall option, I chose to allow only SMTP and SSH traffic. These are the only two services that are required for this server to perform its function, so we want to block all other incoming traffic.

Post-install Configuration:

Once the installation is completed, you need to stop and disable all unneeded services. One of the first things to check is what services are running, and what ports are in use by those services. A handy utility for this is the lsof (list open files) command, with the -Pni options. You can also get a listing of which services are currently enabled by using the chkconfig utility with the --list option, and piping the output through the grep utility (searching for "on").

```
lsof -Pni
chkconfig --list | grep on
```

You can then stop unneeded services with the following command:

```
/etc/rc.d/init.d/<service name> stop
```

These are the services that were stopped on this server:

atd	nfslock	gpm	portmap	isdn
sendmail	lpd	xfs	netfs	xinetd

You can then disable unneeded services with the following command:

```
chkconfig --level 0123456 <service name> off
```

These are the services that were disabled on this server:

atd	nfslock	gpm	portmap	isdn
sendmail	lpd	xfs	netfs	xinetd

As an alternative, you can run the setup utility (/usr/sbin/setup) to accomplish the same thing. It is a graphical tool for enabling/disabling services (and configuring other system parameters).

2. Configure Ipchains

Ipchains is one of the firewall utilities for Linux. It is a derivative of the ipfw utility from BSD, and has gone through many rewrites to get to its current state. Its function is to apply rules to packets as they cross the various input, output, forward, and/or custom chains for an interface. These rules allow packets to be accepted or rejected based on many different criteria such as source address, source port, destination address, destination port, protocol, interface, etc. A good ipchains tutorial is the official ipchains howto, which can be found at:

http://netfilter.samba.org/ipchains/HOWTO.html

With the custom firewall settings that were selected during the Red Hat installation, the following configuration file should have been generated and saved as /etc/sysconfig/ipchains:

```
# Firewall configuration written by lokkit
# Manual customization of this file is not recommended.
# Note: ifup-post will punch the current nameservers through the
#       firewall; such entries will *not* be listed here.
:input ACCEPT
:forward ACCEPT
:output ACCEPT
-A input -s 0/0 -d 0/0 25 -p tcp -y -j ACCEPT
-A input -s 0/0 -d 0/0 22 -p tcp -y -j ACCEPT
-A input -s 0/0 -d 0/0 -i lo -j ACCEPT
-A input -p tcp -s 0/0 -d 0/0 0:1023 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 2049 -y -j REJECT
-A input -p udp -s 0/0 -d 0/0 0:1023 -j REJECT
-A input -p udp -s 0/0 -d 0/0 2049 -j REJECT
-A input -p tcp -s 0/0 -d 0/0 6000:6009 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 7100 -y -j REJECT
```

For the purpose of this mail gateway, the only services that are required are SSH (secure shell) for administration, and SMTP (simple mail transport protocol) for actually sending and receiving email. The script that activates the interface on startup will automatically "punch a hole" in the firewall to all DNS (domain name service) traffic to the server(s) listed in /etc/resolv.conf.

3. Install daemontools

Daemontools is one of many software packages written by D.J. Bernstein. It is a collection of tools used for managing services on a server. The main programs from this package that will be used are svcscanboot, scvscan, supervise, svc, svok, svstat, and multilog. The official descriptions of each program (as quoted from the daemontools site):

- **svscanboot** starts svscan in the /service directory.
- **svscan** starts one supervise process for each subdirectory of the current directory.
- **svc** controls services monitored by supervise.
- **svok** checks whether supervise is running.
- **svstat** prints the status of services monitored by supervise.
- **multilog** reads a sequence of lines from stdin and appends lines to any number of logs.

The installation of the daemontools package is very straight forward. Download the source, unpack it, and run the install script.

```
mkdir -p /package
chmod 1755 /package
cd /package
wget http://cr.yp.to/daemontools/daemontools-0.76.tar.gz
gunzip daemontools-0.76.tar
tar -xpf daemontools-0.76.tar
```

```
rm -f daemontools-0.76.tar
cd admin/daemontools-0.76
package/install
```

The installation script will add the svscanboot command to the /etc/inittab file. This allows it to be automatically started in each runlevel and re-spawned if it crashes.

4. Install ucspi-tcp

The ucspi-tcp package is another software package developed by D.J. Bernstein. It is a collection of TCP utilities and it will be used because it is a faster, more secure replacement for the inetd and/or xinetd daemons. It is used to listen for TCP connections on specified ports, runs chosen programs upon receiving a connection, and can also use access control features to restrict connections. The main programs that will be used from this package are tcpserver, tcprules, and (optionally) rblsmtpd. The official descriptions of each program (as quoted from the ucspi-tcp site):

- **tcpserver** waits for incoming connections and runs a program of your choice.
- **tcprules** compiles rules for tcpserver.
- **rblsmtpd** blocks mail from RBL-listed sites.

The installation of the ucspi-tcp package is very basic. Just download the source, unpack it, and perform a make and make setup check.

```
cd /usr/local/src
wget http://cr.yp.to/ucspi-tcp/ucspi-tcp-0.88.tar.gz
gunzip ucspi-tcp-0.88.tar
tar -xf ucspi-tcp-0.88.tar
cd ucspi-tcp-0.88
make && make setup check
```

5. Install dnsache (djbdns)

The djbdns package is another software package developed by D.J/ Bernstein. It contains several programs designed for DNS name resolution, and has proven to be faster, more secure, and more stable than BIND. Instead of using one big program to handle all possible DNS needs, it uses several smaller programs that can be used only as needed. The installation of the djbdns package is simple, but getting the individual programs to work requires a bit more configuration. To install the package, download the source, unpack it, and perform a make and make setup check.

```
cd /usr/local/src
wget http://cr.yp.to/djbdns/djbdns-1.05.tar.gz
gunzip djbdns-1.05.tar
tar -xf djbdns-1.05.tar
cd djbdns-1.05
make && make setup check
```

After installation is completed, the individual programs that are to be used must be configured and started. This installation will only be using the dnscache program as the local DNS cache for name resolution. Since dnscache runs chrooted, the group and user names must be created.

```
groupadd -r -g 405 djbdns
useradd -d /etc/dnscache -g 405 -u 410 -M -r -s /bin/true dnscache
useradd -d /etc/dnscache -g 405 -u 411 -M -r -s /bin/true dnslog
```

After the necessary group and users are added, dnscache needs to be configured. In this instance, dnscache is being utilized as a caching name server to the localhost only, so it will be configured to listen on the localhost IP address. Run the dnscache-conf program, which will setup the home directory and configure dnscache to run as user dnscache, group dnslog, and listen on 127.0.0.1. Creating the symbolic link to the /etc/dnscache directory in the /service directory will cause the supervise process to execute the /etc/dnscache/run script, which in turn starts dnscache and the dnscache logging process.

```
dnscache-conf dnscache dnslog /etc/dnscache 127.0.0.1
ln -s /etc/dnscache /service
```

The dnscache service should start within 5 seconds of the static link being created. The installation can be tested (if desired) with the dig or nslookup utilities by querying 127.0.0.1 as the server. After dnscache is configured and started, the server needs to be configured to use dnscache for name resolution. This is done by editing the /etc/resolv.conf and adding an entry for nameserver 127.0.0.1.

```
mv /etc/resolv.conf /etc/resolv.conf.old
echo "nameserver 127.0.0.1" > /etc/resolv.conf
```

6. Download & patch Qmail

```
cd /usr/local/src
wget http://cr.yp.to/software/qmail-1.03.tar.gz
tar xzvf qmail-1.03.tar.gz
cd qmail-1.03
```

At this point it is necessary to apply the qmailqueue-patch. Use your favorite editor to create the file "qmailqueue-patch" containing the code found in Appendix A (or downloaded from the qmail.org website). Then simply apply the patch with the following command:

```
patch <qmailqueue-patch
```

You will know it is successful if you see the following messages:

```
patching file Makefile
patching file qmail.c
```

7. Install Qmail

Qmail's installation script will create all of the necessary directories; you only need to create the starting directory.

```
mkdir /var/qmail
```

Note: If you want a more distributed installation, this can be accomplished by creating symbolic links under /var/qmail. For example:

```
mkdir /var/qmail
ln -s /usr/man /var/qmail/man
mkdir /etc/qmail
ln -s /etc/qmail /var/qmail/control
ln -s /usr/sbin /var/qmail/bin
```

Create the users and groups

```
groupadd nofiles
useradd -g nofiles -d /var/qmail/alias alias
useradd -g nofiles -d /var/qmail qmaild
useradd -g nofiles -d /var/qmail qmaill
useradd -g nofiles -d /var/qmail qmailp
groupadd qmail
useradd -g qmail -d /var/qmail qmailq
useradd -g qmail -d /var/qmail qmailr
useradd -g qmail -d /var/qmail qmails
```

NOTE: The above is an example for a Linux system. If you are attempting this on another OS, then refer to the file INSTALL.ids in the Qmail source distribution for OS-specific examples.

Build Qmail

```
make setup check
./config-fast sans.sullys.net     (insert your domain name here)
```

Create startup script /var/qmail/rc

```
#!/bin/sh
# Using stdout for logging
# Using control/defaultdelivery from qmail-local to deliver messages by default

exec env - PATH="/var/qmail/bin:$PATH" \
qmail-start "`cat /var/qmail/control/defaultdelivery`"
```

Then make the script executable and create the qmail log directory::

```
chmod 755 /var/qmail/rc
mkdir /var/log/qmail
```

Setup the default delivery type:

```
echo ./Maildir/ >/var/qmail/control/defaultdelivery
```

Create startup scripts

Link the startup scripts to the different run levels:

```
ln -s /var/qmail/bin/qmailctl /etc/rc.d/init.d/qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc0.d/K30qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc1.d/K30qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc2.d/S80qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc3.d/S80qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc4.d/S80qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc5.d/S80qmail
ln -s /etc/rc.d/init.d/qmail /etc/rc.d/rc6.d/K30qmail
```

Make the qmailctl script executable, and link it to a directory in your path:

```
chmod 755 /var/qmail/bin/qmailctl
ln -s /var/qmail/bin/qmailctl /usr/bin
```

Create the supervise scripts & directories:

```
mkdir -p /var/qmail/supervise/qmail-send/log
mkdir -p /var/qmail/supervise/qmail-smtpd/log
```

Create the /var/qmail/supervise/qmail-send/run file:

```
#!/bin/sh
exec /var/qmail/rc
```

Create the /var/qmail/supervise/qmail-send/log/run file:

```
#!/bin/sh
exec /usr/local/bin/setuidgid qmaill /usr/local/bin/multilog t /var/log/qmail
```

Create the /var/qmail/supervise/qmail-smtpd/run file:

```
#!/bin/sh
QMAILDUID=`id -u qmaild`
```

```
NOFILESGID=`id -g qmaild`
MAXSMTPD=`cat /var/qmail/control/concurrencyincoming`
if [ -z "$QMAILDUID" -o -z "$NOFILESGID" -o -z "$MAXSMTPD" ]; then
  echo QMAILDUID, NOFILESGID, or MAXSMTPD is unset in
  echo /var/qmail/supervise/qmail-smtpd/run
  exit 1
fi
exec /usr/local/bin/softlimit -m 2000000 \
  /usr/local/bin/tcpserver -v -R -l 0 -x /etc/tcp.smtp.cdb -c "$MAXSMTPD" \
  -u "$QMAILDUID" -g "$NOFILESGID" 0 smtp /var/qmail/bin/qmail-smtpd 2>&1
```

Create the concurrencyincoming control file:

```
echo 20 > /var/qmail/control/concurrencyincoming
chmod 644 /var/qmail/control/concurrencyincoming
```

Create the /var/qmail/supervise/qmail-smtpd/log/run file:

```
#!/bin/sh
exec /usr/local/bin/setuidgid qmaill /usr/local/bin/multilog t /var/log/qmail/smtpd
```

Make the run files executable:

```
chmod 755 /var/qmail/supervise/qmail-send/run
chmod 755 /var/qmail/supervise/qmail-send/log/run
chmod 755 /var/qmail/supervise/qmail-smtpd/run
chmod 755 /var/qmail/supervise/qmail-smtpd/log/run
```

Then set up the log directories:

```
mkdir -p /var/log/qmail/smtpd
chown qmaill /var/log/qmail /var/log/qmail/smtpd
```

Finally, link the supervise directories into /service:

```
ln -s /var/qmail/supervise/qmail-send /var/qmail/supervise/qmail-smtpd /service
```

The qmail system will start automatically shortly after these links are created. If you don't want it running yet, do:

```
qmailctl stop
```

Setup SMTP access control

```
echo '127.:allow,RELAYCLIENT="" >>/etc/tcp.smtp
qmailctl cdb
```

You will need to add the IP addresses of all mail servers that you want to relay mail through your gateway server. Qmail checks the IP address of the mail server attempting to relay against its cdb database to determine if relaying is allowed. If your mail server's IP address is 1.2.3.4, then you would need to perform the following:

```
echo '1.2.3.4:allow,RELAYCLIENT="" >>/etc/tcp.smtp
qmailctl cdb
```

This will add your server's IP address and rebuild the cdb database.

Stop and disable the current MTA

Since we started with a clean install, sendmail should be the only MTA installed. We disabled sendmail earlier, now we want to uninstall it, and create links for the old sendmail binaries to point to the Qmail versions:

Uninstall sendmail

```
rpm -e --nodeps sendmail
```

Create links to Qmail binaries

```
mv /usr/lib/sendmail /usr/lib/sendmail.old                  # ignore errors
mv /usr/sbin/sendmail /usr/sbin/sendmail.old                # ignore errors
chmod 0 /usr/lib/sendmail.old /usr/sbin/sendmail.old        # ignore errors
ln -s /var/qmail/bin/sendmail /usr/lib
ln -s /var/qmail/bin/sendmail /usr/sbin
```

Create system aliases:

```
echo "sully@sullys.net" > /var/qmail/alias/.qmail-root
echo "sully@sullys.net" > /var/qmail/alias/.qmail-postmaster
ln -s .qmail-postmaster /var/qmail/alias/.qmail-mailer-daemon
chmod 644 /var/qmail/alias/.qmail-root /var/qmail/alias/.qmail-postmaster
```

NOTE: Replace my email address with the account you want to use.

Start Qmail

```
qmailctl start
```

Test Qmail

```
qmailctl stat
```

Check that we have only the ports we want open listening:

```
[root@sans rc6.d]# lsof -Pni
COMMAND    PID USER   FD   TYPE DEVICE SIZE NODE NAME
sshd       926 root    3u  IPv4   1246       TCP *:22 (LISTEN)
dnscache 14228 root    3u  IPv4  22179       UDP 127.0.0.1:53
dnscache 14228 root    4u  IPv4  22180       TCP 127.0.0.1:53 (LISTEN)
tcpserver 16554 root   3u  IPv4  30148       TCP *:25 (LISTEN)
```

```
ns1:~# nmap -sT -O 66.62.235.150

Starting nmap V. 2.54BETA7 ( www.insecure.org/nmap/ )
Interesting ports on  (66.62.235.150):
(The 1532 ports scanned but not shown below are in state: closed)

Port       State       Service
22/tcp     open        ssh
25/tcp     open        smtp

Nmap run completed -- 1 IP address (1 host up) scanned in 1080 seconds
```

8. Install Qmail-scanner

Qmail-scanner is a Qmail add-on that allows the server to scan all emails for viruses, as well as filter mail based upon specific strings in either mail headers or attachment names. According to the author, Jason Haar, it is integrated into the mail server at a lower level than some other Unix-based virus scanners, resulting in better performance. It is capable of scanning not only locally sent/received Email, but also Email that crosses the server in a relay capacity.

Qmail-scanner relies on several other software packages to function properly. These include the reformime binary from the Maildrop-1.3.6 (or higher) package, the DB_File and Time::HiRes Perl modules,

Download and install Maildrop-1.3.6+

```
wget http://download.sourceforge.net/courier/maildrop-1.3.7.tar.gz
tar xzvf maildrop-1.3.7.tar.gz
cd maildrop-1.3.7
./configure
make
make install-strip
make install-man
```

Verify that DB_File is already installed (it should be by default)

```
[root@sans maildrop-1.3.7]# find / -name DB_File.pm
/usr/lib/perl5/5.6.0/i386-linux/DB_File.pm
```

Get and install Time::HiRes Perl module

```
cd /usr/local/src
wget http://www.cpan.org/authors/id/D/DE/DEWEG/Time-HiRes-01.20.tar.gz
tar xzvf Time-HiRes-01.20.tar.gz
cd Time-HiRes-01.20
perl Makefile.PL
make
make test
make install
```

Download and install TNEF

```
cd /usr/local/src
wget http://world.std.com/~damned/tnef-1.1.1.tar.gz
tar xzvf tnef-1.1.1.tar.gz
cd tnef-1.1.1
./configure
make
make install
```

Download and unpack qmail-scanner

```
cd /usr/local/src
wget http://prdownloads.sourceforge.net/qmail-scanner/qmail-scanner-1.10.tgz
tar xzvf qmail-scanner-1.10.tgz
cd qmail-scanner-1.10
./configure
./configure --install --domain sullys.net
```

Test the installation with the included script

```
./contrib/test_installation.sh -doit
```

You should see the following messages:

Sending standard test message - no viruses... done!
Sending eicar test virus - should be caught by perlscanner module... done!
Sending eicar test virus with altered filename - should only be caught by commercial anti-virus modules (if you have any)... Done!

Once the test is finished, you should go check the email account you setup for administrative purposed to ensure all of the messages were filtered properly.

Set the QMAILQUEUE Variable

Finally, set QMAILQUEUE variable in qmail-smtpd environment and modify the softlimit for memory to around 8 MB. This ensures that all email is passed to the qmail-scanner script, The qmail-scanner script will then pass all "clean" email onto the normal qmail-queue program. It will quarantine any unacceptable email.

```
echo "/var/qmail/bin/qmail-scanner-queue.pl" > /service/smtpd/env/QMAILQUEUE
```

To increase the softlimit setting, edit the /var/qmail/supervise/qmail-smtpd/run file and ensure that the following line exists:

```
exec /usr/local/bin/softlimit -m 8000000
```

This should allocate enough memory to process and scan individual email messages.

9. Install UVScan (or your commercial scanner)

This step of the process is optional, depending on whether or not you have a commercial antivirus product to use with qmail-scanner. It is possible to skip this step entirely and still have an effective scanning tool, but adding a commercial scanner increases the level of security. This demonstrates how to install uvscan from McAfee/NAI.

```
cd /usr/local/src
mkdir uvscan
cd uvscan
wget (your download link for uvscan from your commercial grant account)
./install-uvscan
```

Simply answer the questions that it asks; the default answers should work for you.

```
Which directory do you want to install into? [/usr/local/uvscan]
/usr/local/uvscan doesn't exist. Create it? [y]/n y
Do you want to create the necessary link(s) to uvscan [y]/n y
Do you want to create the necessary link(s) to liblnxfv.so.4 [y]/n y
Do you want to create the necessary link(s) to uvscan.1 [y]/n y
/usr/local/man/man1 doesn't exist. Create it? [y]/n y

Installation complete.

Do you want to perform a scan of all filesystems y/[n] n
[root@sans uvscan]#
```

10. Configure Qmail-scanner

The next step is to configure the perlscanner module in the qmail-scanner script to block attachments that you deem inappropriate, and optionally configure the commercial antivirus support. Edit the /var/spool/qmailscan/quarantine-attachments.txt file and follow the examples contained within for blocking email based on attachment type, subject line, and/or size. The following are examples of different rules:

Blocking by specific subject:

ILOVEYOU Virus-Subject: Love Letter Virus/Trojan

Blocking by specific file:

Happy99.exe 10000 Happy99 Trojan

Blocking by generic file type:

.exe 0 .exe File attachments not allowed

NOTE: The file is a tab-delimited file, so regular spaces will cause errors in configuration.

Appendix E contains a list of basic file extensions that should be blocked. Of course, this will have to be adjusted to meet your individual needs and/or policies. After you edit the file, you need to rebuild the rules database for the scanner. This can be accomplished with the following command:

/var/qmail/bin/qmail-scanner-queue.pl –g

12. Configure Mail Routes

The final step is to edit the MX records for your domain so that the new gateway mail servers are the only mail servers listed and to setup SMTP routes on the gateway servers for final mail delivery. You can setup an SMTP route in qmail by creating a control file called smtproutes (/var/qmail/control/smtproutes) that maps domains to mail servers. The entry would look like the following:

domain.com:mailserver.domain.com (example)
sullys.net:mx1.sullys.net

In the example above, qmail would send all mail destined for the @sullys.net domain to the server mx1.sullys.net. This allows all external mail to be relayed through your gateway servers. You will also need to configure your internal mail servers to use the gateway servers as your SMTP relays in order to scan all outbound email as well.

V. Conclusion

In the case scenario presented here, all inbound and outbound email is now processed by gateway mail servers running qmail with qmail-scanner. Instituting strict attachment filtering rules coupled with a good commercial antivirus product has resulted in no email virus outbreaks since the gateway servers were put into service. It has also greatly reduced the amount of administrative overhead that would be required in keeping multiple email servers updated with all of the latest definition files. There is also the added benefit of having a single chokepoint to cutoff all mail services in the event of a new virus that is not initially handled by qmail-scanner.

References:

1. Life with Qmail
   http://www.lifewithqmail.org/lwq.html

2. Qmail-Scanner
   http://qmail-scanner.sourceforge.net/

3. Qmail-queue patch
   http://www.qmail.org/qmailqueue-patch

4. Qmail: The Internet's MTA of Choice
   http://cr.yp.to/qmail.html

5. CERT® Advisory CA-2001-22 W32/Sircam Malicious Code
   http://www.cert.org/advisories/CA-2001-22.html
   Original release date: July 25, 2001
   Last revised: August 23, 2001

6. CERT® Advisory CA-1999-04 Melissa Macro Virus
   http://www.cert.org/advisories/CA-1999-04.html
   Original issue date: March 27, 1999
   Last revised: March 31, 1999

7. CERT® Advisory CA-2000-04 Love Letter Worm
   http://www.cert.org/advisories/CA-2000-04.html
   Original release date: May 4, 2000
   Last revised: May 9, 2000

8. CERT® Advisory CA-2001-26 Nimda Worm
   http://www.cert.org/advisories/CA-2001-26.html
   Original release date: September 18, 2001
   Revised: September 25, 2001

9. Linux IPCHAINS-HOWTO
   http://netfilter.samba.org/ipchains/HOWTO.html
   Rusty Russell
   v1.0.8, Tue Jul 4 14:20:53 EST 2000

10 . ICSA Labs 6th Annual Computer Virus Prevalance Survey 2000
   http://www.antivirus.com/download/whitepapers/vps20001.pdf

Appendix A: Qmail-queue Patch

May be downloaded with original comments at http://www.qmail.org/qmailqueue-patch

```
diff -u qmail-1.03-orig/Makefile qmail-1.03/Makefile
--- qmail-1.03-orig/Makefile   Mon Jun 15 04:53:16 1998
+++ qmail-1.03/Makefile        Tue Jan 19 10:52:24 1999
@@ -1483,12 +1483,12 @@
 trigger.o fmtqfn.o quote.o now.o readsubdir.o qmail.o date822fmt.o \
 datetime.a case.a ndelay.a getln.a wait.a seek.a fd.a sig.a open.a \
 lock.a stralloc.a alloc.a substdio.a error.a str.a fs.a auto_qmail.o \
-auto_split.o
+auto_split.o env.a
         ./load qmail-send qsutil.o control.o constmap.o newfield.o \
         prioq.o trigger.o fmtqfn.o quote.o now.o readsubdir.o \
         qmail.o date822fmt.o datetime.a case.a ndelay.a getln.a \
         wait.a seek.a fd.a sig.a open.a lock.a stralloc.a alloc.a \
-        substdio.a error.a str.a fs.a auto_qmail.o auto_split.o
+        substdio.a error.a str.a fs.a auto_qmail.o auto_split.o env.a

 qmail-send.0: \
 qmail-send.8

diff -u qmail-1.03-orig/qmail.c qmail-1.03/qmail.c
--- qmail-1.03-orig/qmail.c     Mon Jun 15 04:53:16 1998
+++ qmail-1.03/qmail.c          Tue Jan 19 09:57:36 1999
@@ -6,14 +6,25 @@
 #include "fd.h"
 #include "qmail.h"
 #include "auto_qmail.h"
+#include "env.h"

-static char *binqqargs[2] = { "bin/qmail-queue", 0 } ;
+static char *binqqargs[2] = { 0, 0 } ;
+
+static void setup_qqargs()
+{
+  if(!binqqargs[0])
+    binqqargs[0] = env_get("QMAILQUEUE");
+  if(!binqqargs[0])
+    binqqargs[0] = "bin/qmail-queue";
+}

 int qmail_open(qq)
```

```
 struct qmail *qq;
 {
  int pim[2];
  int pie[2];
+
+  setup_qqargs();

  if (pipe(pim) == -1) return -1;
  if (pipe(pie) == -1) { close(pim[0]); close(pim[1]); return -1; }
```

Appendix B: Supported Antivirus Scanners

The following virus scanners are known to work with qmail-scanner:

Trend's Virus scanner
http://www.antivirus.com/

Sophos's "sweep" virus scanner
http://www.sophos.com/

H+BEDV's antivir scanner
http://www.hbedv.com/

Kaspersky's AVPLinux scanner
http://www.kaspersky.com/

MacAfee's (NAI's) virus scanner
http://www.nai.com

Command's virus scanner
http://www.commandcom.com/

F-Secure Anti-Virus scanner
http://f-secure.com/

F-Prot Anti-Virus scanner
http://www.f-prot.com/

InocuLAN Anti-Virus scanner
http://www.cai.com/

Sophie: Daemon front-end to Sophos Sweep
http://www.vanja.com/tools/

Trophie: Daemon front-end to Trend iscan
http://www.vanja.com/tools/

The following is not a antivirus product, but instead a SPAM filter that works with Qmail-scanner:

Spam Assassin Daemon
http://spamassassin.taint.org/

Appendix C: Qmail Startup Script

```
#!/bin/sh

# For Red Hat chkconfig
# chkconfig: - 80 30
# description: the qmail MTA

PATH=/var/qmail/bin:/bin:/usr/bin:/usr/local/bin:/usr/local/sbin
export PATH

QMAILDUID=`id -u qmaild`
NOFILESGID=`id -g qmaild`

case "$1" in
  start)
    echo "Starting qmail"
    if svok /service/qmail-send ; then
      svc -u /service/qmail-send
    else
      echo qmail-send supervise not running
    fi
    if svok /service/qmail-smtpd ; then
      svc -u /service/qmail-smtpd
    else
      echo qmail-smtpd supervise not running
    fi
    if [ -d /var/lock/subsys ]; then
      touch /var/lock/subsys/qmail
    fi
    ;;
  stop)
    echo "Stopping qmail..."
    echo "  qmail-smtpd"
    svc -d /service/qmail-smtpd
    echo "  qmail-send"
    svc -d /service/qmail-send
    if [ -f /var/lock/subsys/qmail ]; then
      rm /var/lock/subsys/qmail
    fi
    ;;
  stat)
    svstat /service/qmail-send
    svstat /service/qmail-send/log
```

```
  svstat /service/qmail-smtpd
  svstat /service/qmail-smtpd/log
  qmail-qstat
  ;;
doqueue|alrm|flush)
  echo "Flushing timeout table and sending ALRM signal to qmail-send."
  /var/qmail/bin/qmail-tcpok
  svc -a /service/qmail-send
  ;;
queue)
  qmail-qstat
  qmail-qread
  ;;
reload|hup)
  echo "Sending HUP signal to qmail-send."
  svc -h /service/qmail-send
  ;;
pause)
  echo "Pausing qmail-send"
  svc -p /service/qmail-send
  echo "Pausing qmail-smtpd"
  svc -p /service/qmail-smtpd
  ;;
cont)
  echo "Continuing qmail-send"
  svc -c /service/qmail-send
  echo "Continuing qmail-smtpd"
  svc -c /service/qmail-smtpd
  ;;
restart)
  echo "Restarting qmail:"
  echo "* Stopping qmail-smtpd."
  svc -d /service/qmail-smtpd
  echo "* Sending qmail-send SIGTERM and restarting."
  svc -t /service/qmail-send
  echo "* Restarting qmail-smtpd."
  svc -u /service/qmail-smtpd
  ;;
cdb)
  tcprules /etc/tcp.smtp.cdb /etc/tcp.smtp.tmp < /etc/tcp.smtp
  chmod 644 /etc/tcp.smtp.cdb
  echo "Reloaded /etc/tcp.smtp."
  ;;
help)
  cat <<HELP
 stop -- stops mail service (smtp connections refused, nothing goes out)
```

```
  start -- starts mail service (smtp connection accepted, mail can go out)
  pause -- temporarily stops mail service (connections accepted, nothing leaves)
   cont -- continues paused mail service
   stat -- displays status of mail service
    cdb -- rebuild the tcpserver cdb file for smtp
restart -- stops and restarts smtp, sends qmail-send a TERM & restarts it
doqueue -- schedules queued messages for immediate delivery
 reload -- sends qmail-send HUP, rereading locals and virtualdomains
  queue -- shows status of queue
   alrm -- same as doqueue
  flush -- same as doqueue
    hup -- same as reload
HELP
    ;;
  *)
    echo "Usage: $0 {start|stop|restart|doqueue|flush|reload|stat|pause|cont|cdb|queue|help}"
    exit 1
    ;;
esac

exit 0
```

Appendix D: Software Packages and Patches

Please note that these links are to the homepage for the software, and are not direct downloads.

Official Sites

Red Hat 7.2
http://www.redhat.com/

Daemontools
http://cr.yp.to/daemontools.html

UCSPI-TCP
http://cr.yp.to/ucspi-tcp.html

DJBDNS
http://cr.yp.to/djbdns.html

Qmail 1.03
http://cr.yp.to/qmail.html

Qmail-queue Patch
http://www.qmail.org/qmailqueue-patch

Maildrop 1.3.6+ (reformime)
http://download.sourceforge.net/courier/

Perl 5.005_03+
http://www.perl.com/

Perl module Time::HiRes
http://search.cpan.org/search?module=Time::HiRes

Perl module DB_File
http://search.cpan.org/search?module=DB_File

TNEF 1.1.1
http://world.std.com/~damned/software.html

Sullys.net Mirror

http://www.sullys.net/sansproject/

Appendix E: Recommended Attachments to Block

.exe	0	Executable binary
.com	0	Non relocable MSDOS executable binary
.vbs	0	Visual Basic Script
.vba	0	Visual Basic Application
.shs	0	Shell automation code
.scr	0	Screen Saver
.bat	0	COMMAND.COM batch file
.btm	0	JP Software fast batch file
.reg	0	Windows Registry file
.msi	0	Executable binary
.msc	0	Executable binary
.chm	0	Compiled HTML help file
.inf	0	Windows INF file
.cpl	0	Control Panel library
.wsf	0	Windows Scripting File
.vbe	0	VisualBasic Encoded
.js	0	JavaScript
.jse	0	JavaScript Encoded
.css	0	Cascading Style Sheets
.wsh	0	Windows Scripting Host
.sct	0	Scriptlet File
.hta	0	HTML Application
.lnk	0	Windows Explorer links
.cmd	0	cmd.exe NT batch
.pif	0	Windows Program Information Files
.ade	0	Access Project Extension
.mdb	0	Access Application
.adp	0	Access Project
.mde	0	Access MDE Database
.bas	0	Visual Basic Class Module
.msp	0	Windows Installer Patch
.mst	0	Visual Test Source File
.pcd	0	Photo CD Image
.crt	0	Security Certificate
.hlp	0	Windows Help File
.url	0	Internet Shortcut (Uniform Resource Locator)
.ins	0	Internet Communication Settings
.isp	0	Internet Communication Settings
.vb	0	Script Script File
.wsc	0	Windows Script Componen
.shb	0	Shell Scrap object

Last Updated: May 28th, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event