



# Q-Mail Server in the Enterprise

Use the step-by-step instructions given here to implement a Q-Mail server in your organisation. It will take only a few hours to do so.

**T**he mail server discussed here is designed on the enterprise concept and based on the Q-Mail server. Q-Mail implementation on the Linux platform is not only very secure and fast but also very scalable. The Q-Mail implementation will cater to the following aspects of an enterprise:

a. In this implementation, you will see how Q-Mail can be implemented along with non-system users. Q-Mail implementation allows the use of e-mail without being users of the system, thereby minimising the security threat posed by system users. The implementation allows easy user management through a Web-based user interface.

b. Q-Mail allows multiple domains pointing to the same IP, an essential requirement for an enterprise environment.

c. Q-Mail allows spam control, a critical aspect in an enterprise, through various methods discussed later in the article.

d. Q-Mail allows implementation of log analysers and mrtg graphs that help the administrator to upgrade, tune and take corrective actions to suit the changing environment. Traffic analysis is done on factors such as load factor, speed of the mail server,

maximum load time schedule, potential users of the mail server and daily/weekly/monthly traffic.

e. As this implementation is on the Linux platform, you can build firewalls using ipchains/iptables to suit the environment.

f. Virus protection on Q-Mail can be done through free/commercial Virus Walls available, which will be discussed later in this article.

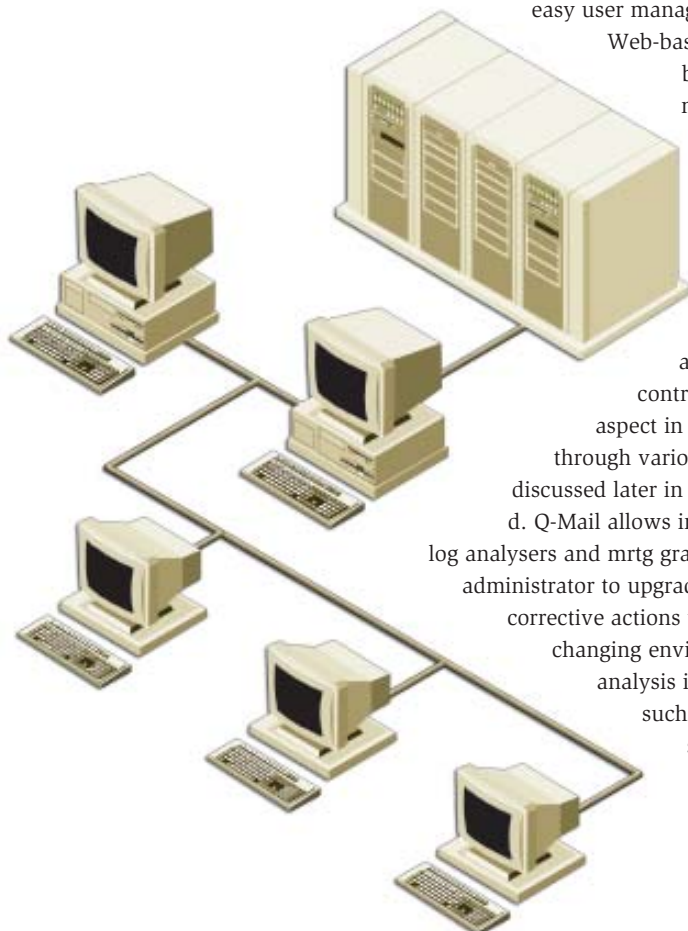
g. Some of the other aspects that the administrator is generally worried about are deployment time, ease of usage, ease of administration. Web-based tools such as qmailadmin can be used to take care of deployment issues such as user administration, specific quota allocation (to limit the mail box), vacation admin, password administration, creation and management of mailing lists and many more.

h. Another crucial issue in corporate environments is implementation of local DNS, allowing administrators to have better control over DNS rather than third party DNSs. The DNS server is sometimes preferred because of the changing environment of domains/sub-domains in corporates (to be covered in our next issue).

## INSTALLATION PLATFORM

Next, we will choose the installation platform for the implementation. Since the solution here is built around open source, you can choose a flavour of Linux you prefer. We have chosen RedHat 7.1 with kernel 2.4.2-2 on a multi-processor server, but it should work equally well with any other flavour of Linux.

The choice of hardware is left to you, which will mainly depend upon the type of environment you are working in. The RedHat Linux 7.1 custom/full installation was carried



out. The firewall was disabled and activated at a later stage. You can also go for server-based installation and later on add required components as and when required. This server was deployed for the corporate Intranet so we had two LAN cards—one with one global IP and the other with a local IP.

In case you are planning to co-locate the server at the ISP or deploying the ISP mail server, you can opt for one LAN card with global IP mapped. If you happen to choose any other flavour of Linux other than Red Hat, please make sure you have upgraded kernel 2.4x to support IPtables. In our case, the Fully Qualified Domain Name (FQDN) is mail.linuxforyou.com.

Please also note that the home directory should be allocated maximum space as all the mailboxes will finally be created on /home directory. Apart from /home directory, /var and /usr will also need sufficient amount of disk space to hold logs and installed packages. Care should be taken while installation, as changing and migration can be a pain later on.

## Q-MAIL INSTALLATION

Before starting the Q-Mail installation, let's take a quick look at the history and implementation of Q-Mail. Q-Mail is a modern replacement for Sendmail, written by Dan Bernstein. Q-Mail is proven to be more secure than Sendmail, and much faster. A number of large Internet sites use Q-Mail. Hotmail's outgoing mail (although Microsoft says it's going to transition to W2K), USA.net's outgoing email, Address.com, Rediffmail.com, Colonize.com, Yahoo! mail, Network Solutions, Verio, MessageLabs (searching 20M e-mails/week for mailware), Listserv.acsu.buffalo.edu (a big listserv hub, using Q-Mail since 1996), Ohio State (biggest US University), Onelist.com (which has merged with e-groups, another big free mailing list service), Listbot, USWest.net (Western US ISP), RIPE, Telenordia, Gmx.de (German ISP), NetZero (free ISP), Critical Path (e-mail outsourcing service with 15 million mailboxes), PayPal/Confinity, Hypermart.net, Casema, Pair Networks, Topica, MyNet.com.tr, FSmal.net, and Vuurwerk.nl.

Q-Mail as a big picture is quite a difficult subject to understand and to implement. But there is multiple implementation and research, which made the big and complicated picture of Q-Mail quite simple for specific environments. Our implementation will be around a specific enterprise environment with a large number of users in multiple domains. Traffic on this mail server can vary from 10,000 to 1 million mails per day. Users can use POP3/IMAP as well as Web protocols to retrieve and send mails. The authentication methods are independent of the Unix/Linux environments, which means that users are not the system's users unlike that of Exchange Server, which are generally a part of the Windows NT domain. Administration is Web-based and mail users can change their profile via the Web browser.

We will implement Q-Mail with a package given along with this issue of LINUX For You. The package is Qinstall—Qinstall is a Q-Mail installer script that allows you to quickly and easily install Q-Mail and other useful packets (daemontools, ucspi-tcp, vpopmail, qmailadmin, ezmlm, sqwebmail). It has a GUI and multi-language support. You can also download this from <http://obua.org/>. Let us try to understand the packages, their integration with Q-Mail and the functionality given by the qinstall script.

**Q-Mail:** Basic mail server with MTA/MDA and POP server.

**Daemon tools:** Collection of tools for managing Unix services. Q-Mail mainly uses services and multilog tools to make sure the service is not hung up and logging the data for Q-Mail respectively.

**Ucspi-tcp:** Earlier, Unix used inetd, which had many security holes. This has been replaced by tools supplied by ucspi-tcp. It consists of tcpserver and tcpclient, which are easy-to-use command-line tools for building TCP client-server applications.

**Vpopmail (vchkpw):** This is a collection of programs and a library to automate the creation and maintenance of virtual domain e-mail configurations for Q-Mail installations using either a single UID/GID or any valid UID/GID in /etc/passwd with a home directory. Features are provided in the library for other applications, which need to maintain virtual domain e-mail accounts. It supports named or IP-based domains. It works with vqadmin, qmailadmin, vqregister, sqwebmail, and courier-imap. It supports MySQL, Sybase, Oracle, LDAP, and file-based (DJB constant database) authentication. It supports SMTP authentication combined with the qmail-smtp-auth patch. It supports user quotas and roaming users (SMTP relay after POP authentication).

**Qmailadmin:** Web control panel for the administration of qmail/vpopmail-based POP, IMAP, LDAP, or Web mail accounts, forwarding, aliases, autoresponders, and mailing lists. It is perfect for ISPs, Web hosting sites, or companies that want to provide a good interface for managing e-mail accounts. It works with Q-Mail, vpopmail, sqwebmail, and courier-imap.

**Ezmlm:** This lets users set up their own mailing lists within Q-Mail's address hierarchy.

**SqWebMail:** SqWebMail is a Web CGI client for sending and receiving e-mail using Maildir mailboxes. The features are very lightweight. It reads mail directly from maildirs. Many (but not all) display elements can be customised without changing the program code. It has hierarchical mail folders and shared folders. Vpopmail and LDAP authentication is available with address book support. It can import e-mail addresses from external LDAP address books. It displays HTML messages (optional). It has extensive MIME support with MIME flowed text format, and delivery status notifications are recognised and nicely formatted. Spell check is available, if you have ispell installed. It can optionally use

gzip compression to return some large Web pages, if a modern browser is used that supports gzip compression.

**Autoresponder:** This facilitates writing vacation admin as well as autoresponding messages for individual mailboxes.

And now, let's get down to business. Please make sure that you have Python installed. As part of the full installation of Red Hat Linux, Python gets installed automatically. Check it with `rpm-q python`. If you get results, that is fine. Otherwise, you have to get it and install it. In case of Red Hat it can be obtained at

```
ftp://rpmfind.net/linux/redhat/updates/7.1/en/os/i386/python-1.5.2-42.71.i386.rpm
```

Once you are through with python installation, download the `qinstall-1.1.tgz` (supplied with CD) in `/tmp` directory. Move to `tmp` directory and untar the package by entering

```
#tar xvfz qinstall-1.1.tgz
```

Modify the file `/tmp/qinstall-1.1/qinstall.conf` enable `_rblsmtpd,"no"` to enable `_rblsmtpd,"yes"`. This is done to have spam protection, which while sending and receiving mails, checks against databases containing spam-related information. Changes are also required in case your document root directory of the Apache Web server is different from `/var/www/html`. Please modify the file accordingly.

```
#cd /tmp
#./qinstall -gui ( it is better this way )
```

This will do the following for you

1. Unpack the sources in `/usr/local/src`.
2. Clean the system from previous installations.
3. Install Q-Mail in `/var/qmail`, create `/etc/init.d/qmail` script. And remove Sendmail from the system.
4. Install `ucspi-tcp`.
5. Patch `daemontools` for Red Hat (`tai64nlocal.c` file) and install it.
6. Install `vpopmail` in the directory that is defined in the configuration section (default is `/home/vpopmail`). `vpopmail` is configured with—`enable-roaming-users = y`.
7. Install `autorespond`.
8. Install `ezmlm` with `idx` patch.
9. Install `qmailadmin`.
10. Install `Sqwebmail`

See how simple it is! This script is tested on Red Hat Linux, Gelecek Linux, Mandrake Linux, Slackware Linux, Yellowdog Linux, Debian Linux and FreeBSD (different script).

Now, we come to some tuning parts. First, Q-Mail needs to be restarted by itself when the machine reboots, so run `ntsysv` to activate Q-Mail at the time of booting. As a default policy, Q-Mail gets installed in `/var` directory. You have to assign a postmaster to which the root's or mailer-daemon's

Control	Default	Used by	Purpose
<code>badmailfrom</code>	none	<code>qmail-smtpd</code>	blacklisted From addresses
<code>concurrencyincoming</code>	20	<code>/service/qmail-smtpd/run</code>	max simultaneous incoming SMTP connections
<code>concurrencylocal</code>	10	<code>qmail-send</code>	max simultaneous local deliveries
<code>concurrencyremote</code>	20	<code>qmail-send</code>	max simultaneous remote deliveries
<code>defaultdelivery</code>	none	<code>/var/qmail/rc</code>	default .qmail file
<code>defaultdomain</code>	me	<code>qmail-inject</code>	default domain name
<code>databytes</code>	0	<code>qmail-smtpd</code>	max number of bytes in message (0=no limit)
<code>locals</code>	me	<code>qmail-send</code>	domains that we deliver locally
<code>me</code>	FQDN of system	various	default for many control files
<code>rcpthosts</code>	none	<code>qmail-smtpd</code>	domains that we accept mail for
<code>smtpgreeting</code>	me	<code>qmail-smtpd</code>	SMTP greeting message
<code>smtproutes</code>	none	<code>qmail-remote</code>	artificial SMTP routes
<code>virtualdomains</code>	none	<code>qmail-send</code>	virtual domains and users

mail will be addressed. There are three files:

`/var/qmail/aliases/.qmail-root`, `/var/qmail/aliases/.qmail-postmaster` and `/var/qmail/aliases/.qmail-mailer-daemon`. If these are not there, please create them with an entry like `&admin@linuxforyou.com` in the first line. This means any mail to such an address will be rerouted to the proper mailbox named `admin@linuxforyou.com`. Your Q-Mail has a control directory under `/var/qmail` which has control functions as described below.

At present, the `concurrency incoming` is important. It controls the `smtp` concurrency at a time, which by default is set to 20 but can be increased till 200 depending on the load on the `smtp`, beyond which recompilation of Q-Mail is needed. The next default delivery is set to `./Maildir/`, which is proprietary of Q-Mail. `Rcpthosts` is responsible for which domains Q-Mail will receive mails for. It is very important to have this file as it is also responsible for anti-spam activities (this is explained later).

The rest are common files as given below:

`/var/qmail/bin`—directory containing Q-Mail executables

`/var/qmail/supervise`—directory linked to `/service` and responsible for running daemon for `smtp`, `pop` and `log`

`/var/log/qmail`—directory containing logs of Q-Mail

`/var/qmail/users/assign`—file containing the virtual domain and related paths

`/var/qmail/queue/`—directory containing all the queues, i.e., `local`, `remote`, `bad`, `bounce`, `to do`, etc

`/var/qmail/doc`—directory that contains documentation

You can stop/start/restart the Q-Mail services

```
# service qmail
{start|stop|restart|doqueue|flush|reload|stat|pause|cont|cdb|queue|help}
```



```
</body>
</html>
```

Now the URL `http://yourserverip/` will take them to the Web interface of Sqwebmail. The user can proceed to login with `userid@domainname` and the password supplied to him.



Administrators can modify the look and feel of the first page of Sqwebmail towards a more corporate

impression by changing `/usr/local/share/sqwebmail3/html/en/index.html` and `invalid.html` and copy images to `/usr/local/share/sqwebmail3/html/en/images` folder.

## IMAP ASPECT OF Q-MAIL—COURIER IMAP

Q-Mail installation is tested and ready to be used as POP/Web protocols. Some sites seriously lack the implementation of an IMAP server. The implementation of an IMAP has an added advantage over POP3 protocol. As IMAP is a protocol, which is authentication- and mailbox-protocol based, you need to have an IMAP server, which can support vchkw (authentication method of VPOPMAIL) and Maildir format of Q-Mail. Courier Imap server not only supports both the specified environments but is also a very reliable, fast and scalable server. The Courier IMAP server has several modules provided to authenticate using the traditional password/shadow files, via the PAM library, from a table on a MySQL server, or from an LDAP server (which requires MySQL or OpenLDAP). An experimental authentication module for PostgreSQL is also available. It also supports OpenSSL.

Download Courier-IMAP from <http://prdownloads.sourceforge.net/courier/courier-imap-1.7.0.tar.bz2> or from the LFY CD to `/tmp`.

```
#cd /usr/local/src/
#bunzip /tmp/courier-imap-1.7.0.tar.bz2
#tar xvf courier-imap-1.7.0.tar
#cd courier-imap-1.7.0
configuring may take some time...
#./configure --disable-root-check --without-authdaemon --with-
authvchkw \
--enable-workarounds-for-imap-client-bugs
#make
#make install-strip
#make install-configure
#cp courier-imap.sysvinit /etc/rc.d/init.d/courier-imap
#chmod 755 /etc/rc.d/init.d/courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc0.d/K30courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc1.d/K30courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc2.d/S80courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc3.d/S80courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc5.d/S80courier-imap
#ln -s ../init.d/courier-imap /etc/rc.d/rc6.d/K30courier-imap

Edit /usr/lib/courier-imap/etc/imapd
Change 'AUTHMODULES="..."' to 'AUTHMODULES="authvchkw"'
Change 'IMAPDSTART=NO' to 'IMAPDSTART=YES'
```

To run courier-imap as `vpopmail.vchkw`

```
Edit /usr/lib/courier-imap/libexec/imapd.rc
Change:
/usr/lib/courier-imap/libexec/couriertcpd -address=$ADDRESS \
```

To:

```
/usr/lib/courier-imap/libexec/couriertcpd -address=$ADDRESS \
-user=vpopmail -group=vchkw \

Start IMAP server
# /etc/rc.d/init.d/courier-imap start
Test the IMAP server ...
#telnet 0 143
Trying 0.0.0.0...
Connected to 0 (0.0.0.0).
Escape character is '^]'.
* OK Courier-IMAP ready. Copyright 1998-2002 Double Precision,
Inc. See COPYING for distribution information.
```

Configure a client (Outlook Express/Outlook2000) to fetch mail via IMAP to test its working.

## RELAY CONCEPT AND IMPLEMENTATION

Let's try to understand what exactly relaying is, to what extent it can be harmful and how it can be prevented? You've your Q-Mail server up and running. It's hosting a few domains. You've set it up such that the Q-Mail can take connections on port 25 to receive mail from other hosts. Another host on the Internet connects to your server on port 25. This might be another mail server running Q-Mail, Sendmail, or some other mail transfer agent, or it might be an end user, who wants to send mail from his desktop mail client. The SMTP conversation starts off with the remote host identifying itself:

```
HELO somehost.somewhere.net
Your server responds:
250 mail.linuxforyou.com
The remote host sends the 'From' part of the envelope:
MAIL FROM:someone@somewhere.net
Your host responds with:
250 ok
The remote host now sends one or more RCPT TO commands, which
specify the recipients of the message:
RCPT TO:someone@elsewhere.com
Just a minute! elsewhere.com is not one of the domains that
your server hosts. What does your server do? It can agree to
accept the message and attempt to deliver it:
250 ok
Or it can reject it:
553 sorry, that domain isn't in my list of allowed rcpthosts
(#5.7.1)
```

In the first case, your server is acting as a relay: it's accepting and agreeing to try to deliver a message that's not destined for a domain that your server hosts. In the second case, it's refusing to act as a relay.

Q-Mail's `rcpthosts` file, which gets its name from the RCPT TO command, determines whether the recipient will be accepted: it will be accepted if and only if the domain of the address given in the RCPT TO command is listed in `rcpthosts`. (Exceptions to this rule will be discussed later on.) If the

rcpthosts file does not exist, then any recipient will be accepted. An SMTP server is an ‘open relay’ if it agrees to relay mail no matter who is sending it—if another host connects to port 25 with some mail, an open relay will accept and try to deliver it no matter what its destination is and who is sending it. A Q-Mail server without a rcpthosts file will act as an open relay.

Why is open relay dangerous? Spammers, the mass-mailers of unsolicited commercial e-mail, can use this open relay to flood with mails, which consume bandwidth and server resources. The administrators of the relaying server are likely to be flooded with complaints from spam recipients. The relaying server may even find itself blacklisted, so that other hosts on the Internet will refuse to receive mail from it (see <http://www.ordb.org>, for example). Leaving your mail relay open these days is considered being irresponsible.

To fix the open relay problem, list your domains in rcpthosts file (/var/qmail/control/rcpthosts). This will only allow the mails generated from listed domains to be relayed.

Our implementation is allowed to have the relay function activated for connections coming from any IP (we have

activated vpopmail with “roaming user = yes”), which means anybody can use our server as relay. The implementation takes care of this selective relaying by implementing POP authentication before SMTP. That means when any client connects to the mail server, it needs to pop its mails before sending any mail across this server.

If you happen to compile the server without this feature then you can allow selective IP address/network (can be a typical scenario in an Intranet) by adding

```
192.168.10.:allow,RELAYCLIENT=""
:allow

in /home/vpopmail/etc/tcp.smtp (where 192.168.10 is your local network segment)
```

and recreate the database file by

```
#tcprules /home/vpopmail/etc/tcp.smtp.cdb /home/vpopmail/etc/
tcp.smtp.temp < /home/vpopmail/etc/tcp.smtp
```

You can reload the database by

```
#service qmail cdb
```

Only clients belonging to the network mentioned will be allowed to relay.

You can check whether your relay is open or not at sites such as <http://spamcop.net>, <http://ordb.org> just to confirm your implementation.

## Spam prevention & implementation

Let's try to understand what spam is and how to handle it. Spam is defined here as unsolicited commercial e-mail, usually sent in bulk. In other words, spam is simply electronic junk mail. Dealing with spam is, at best, a very difficult task. This is mostly true because spammers have a wide array of tools and circumstances available to them that make it easy for them to send you mail but difficult for you to communicate with them or have any authority over them.

Spam is also difficult to deal with because it almost always comes in the guise of a normal e-mail message. No amount of technology can automatically decide what content is undesirable to you, but there are many ways to use technology to reduce the amount of unwanted e-mail you or your users receive.

Anti relay or selective relay must have given you some idea of controlling spam that tries to use your mail server as relay.

One simple and effective way is to identify the e-mail address, network or domain, which is responsible for causing spam and can be mentioned in the control file, i.e., *badmailfrom* (/var/qmail/control/badmailfrom).

While receiving mail from and sending to our mail server, check certain databases such as [ordb.org](http://ordb.org) and [mail-abuse.org](http://mail-abuse.org). Check whether the mail server from which it is receiving or sending to is listed on these databases and only then carry out the required activity. This feature is enabled via “rblsmtp” compiled Q-Mail (true in our case).

So we are more or less safe in terms of spam control. But spam keeps changing and the administrator should be geared to tackle such situations.

Some anti-spam based packages are listed below:

**Efilter:** <http://www.inter7.com/eps/efilter-0.2.tar.gz>

**Bayesbam:** <http://www.garyarnold.com/projects.php#bayesbam>

**BlackMail:** <http://www.jsm-net.demon.co.uk/blackmail/source/>

**EnderUNIX spamGuard:** <http://www.enderunix.org/spamguard/>

**OSpam:** <http://omail.omnis.ch/ospam/>

**Q-Mail Spam Filter:** <http://prinzess.dyndns.org>

**Qmfilt:** <http://sourceforge.net/projects/qmfilt/>

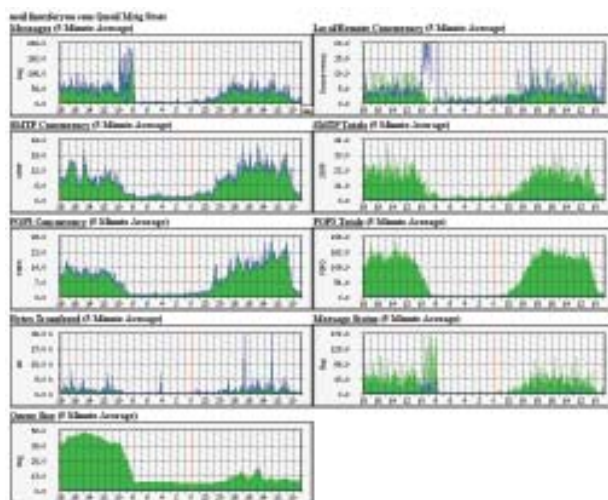
**QSpamDog:** <http://www.dajudge.homeip.net/proj/smalldev/qspamdog-0.1.tar.gz>

**Spamfilter:** <http://www.algonet.se/~staham/linux/programs.html>

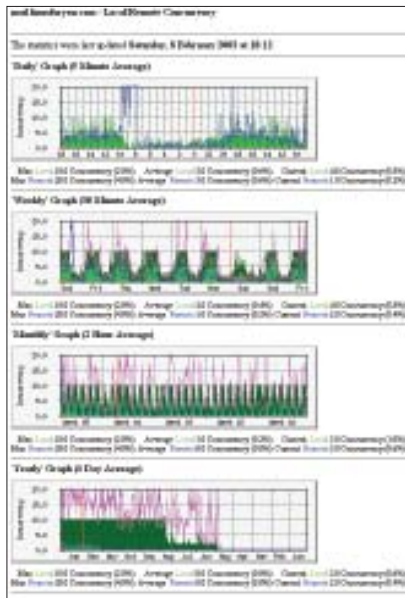
**SpamRule:** <http://isle.wumpus.org/cgi-bin/pikie?SpamRule>

## QMAIL-MRTG FOR ANALYSIS OF TRAFFIC

Once Mailserv is fully functional and has got used to its limits, it is very important to know how effectively your mail server is getting used and to see if anything that's not expected is happening or not. These kinds of queries can be addressed by the implementers or the administrator by implementing qmail-Mrtg. The Multi Router Traffic Grapher (MRTG) is a tool that monitors the traffic load on network-



Qmail MRTG statistics



MRTG graph for local-remote concurrency

processes the logs (large sites with historical logs of over 100 MB can be processed in a few seconds).

For Q-Mail, it graphs remote/local delivery concurrency, queue size, messages process, bytes transferred, and success/failure delivery status. For POP and SMTP, it graphs total connections and concurrencies. `qmailmrtg7` takes the `pop3` `smtp` and Q-Mail transaction logs and sends them to nine different mrtg graphs, where each graph has 4 historical time series. You can see the screen shots.

To install `qmailmrtg7` you need to have the `mrtg` package, which can be downloaded from the supplied CD or from the site <http://people.ee.ethz.ch/~oetiker/webtools/mrtg/pub/mrtg-2.9.14.tar.gz>

In order to compile and use `mrtg`, you need a C compiler and a copy of Perl installed on your machine, which you already have since you have done a full installation of Red Hat linux. Additionally, you must have the `gd`, `zlib` and `libpng` packages and install them as below.

```
# cd /usr/local/src
If you do not have zlib installed:
# wget http://www.gzip.org/zlib/zlib-1.1.4.tar.gz
#gunzip -c zlib.tar.gz | tar xf -
#mv zlib-?.?./ zlib
#cd zlib
#./configure
# make
# cd ..
```

If you don't have `libpng` installed:

```
#wget http://www.libpng.org/pub/png/src/libpng-1.0.12.tar.gz
#gunzip -c libpng-*.tar.gz |tar xf -
#rm libpng-*.tar.gz
#mv libpng-* libpng
#cd libpng
#make -f scripts/makefile.std CC=gcc ZLIBLIB=./zlib
```

links. MRTG generates HTML pages containing graphical images, which provide a 'live' visual representation of this traffic. MRTG is based on Perl and C, and works under Unix. `qmailmrtg7` is a collection of a script, which utilises Q-Mail and `tcpserver/multilog's` extensive logging capabilities to create mrtg graphs. It efficiently

```
ZLIBINC=./zlib
#rm *.so.* *.so
#cd ..
And now you can compile gd
#wget http://www.boutell.com/gd/http/gd-1.8.3.tar.gz
#gunzip -c gd-1.8.3.tar.gz |tar xf -
#mv gd-1.8.3 gd
#cd gd
```

The `\` characters at the end of the following lines mean that all the following material should actually be written on a single line.

```
#make INCLUDEDIRS="-I. -I../zlib -I../libpng" \
LIBDIRS="-L../zlib -L- -L../libpng" \
LIBS="-lgd -lpng -lz -lm"
# cd ..
```

Now everything is ready for the `mrtg` compilation.

```
#cd /usr/local/src
#gunzip -c mrtg-2.9.14.tar.gz | tar xvf -
#cd mrtg-2.9.14
```

If all the libraries have been preinstalled on your system, you can configure `mrtg` by doing a simple:

```
#!/configure --prefix=/usr/local/mrtg-2
```

Otherwise, you may have to give some hints on where to find the various libraries required to compile `mrtg`:

```
#!/configure --prefix=/usr/local/mrtg-2 \
-with-gd=/usr/local/src/gd \
-with-z=/usr/local/src/zlib \
-with-png=/usr/local/src/libpng
```

`Configure` will make sure your environment is fit for building `mrtg`. If it finds a problem, it will tell you so and it will also tell you what to do about it. If everything is fine, you will end up with a custom Makefile for your system. Now type:

```
# make
#make install
```

All the software required by `mrtg` is now installed under the `/usr/local/mrtg-2` subdirectory.

`Mrtg` implementation can also be used to generate traffic graphs from your SNMP-enabled network components, such as routers and switches. But that is not a point of discussion at present. Now we are ready to install `qmailmrtg7`. Please download the package in `/usr/local/src` directory from the CD or from the site

<http://www.inter7.com/qmailmrtg7/qmailmrtg7-3.4.tar.gz>

One more point—make sure your Apache server (`httpd`) is configured and running.

```
#cd /usr/local/src; tar xvfz qmailmrtg7-3.4.tar.gz
#cd qmailmrtg7-3.4
#make ;make install
All these will copy the executable in /usr/local/bin directory
```



```

#/etc/rc.d/init.d/ipchains restart
Testing of opened port on the Mailserver can be scanned by
# nmap -v -sU -sS localhost
This should give you the following result
Starting nmap V. 2.54BETA7 ( www.insecure.org/nmap/ )
Host localhost (127.0.0.1) appears to be up ... good.
Initiating SYN Stealth Scan against localhost (127.0.0.1)
Adding TCP port 80 (state open).
Adding TCP port 22 (state open).
Adding TCP port 143 (state open).
Adding TCP port 25 (state open).
Adding TCP port 110 (state open).
Adding TCP port 53 (state open).
Adding TCP port 1812 (state open).
The SYN Stealth Scan took 4 seconds to scan 1553 ports.
Initiating UDP Scan against localhost (127.0.0.1)
The UDP Scan took 5 seconds to scan 1553 ports.
Interesting ports on localhost (127.0.0.1):
(The 3094 ports scanned but not shown below are in state:
closed)
Port      State      Service
22/tcp    open       ssh
25/tcp    open       smtp
53/tcp    open       domain
53/udp    open       domain
80/tcp    open       http
110/tcp   open       pop-3
143/tcp   open       imap2
1812/tcp  open       unknown
Nmap run completed-1 IP address (1 host up) scanned in 9
seconds

```

This result also points out any of the ports that may be open on the mail server, apart from the required ones.

### Q-MAIL ANTI-VIRUS WALL

Normally, an e-mail virus comes as an attachment, so a corporate policy designed to receive and deny specific attachments reduces the chances of getting viruses as e-mail attachments. This can be the first step to protect the server from malicious virus attacks via e-mails. A script checkattach is supplied with the CD, which can be downloaded in the /home/vpopmail/bin directory

```
# chmod 555 /home/vpopmail/bin/checkattach
```

Modify the type of attachment you would not like to receive on any mailbox and modify the /home/vpopmail/domains/domain\_name/.qmail-default (where domain\_name is the domain as in our case linuxforyou.com) for all the domains mapped on this server as

```

/home/vpopmail/bin/checkattach
| /home/vpopmail/bin/vdelivermail '' delete

```

This will start protecting the mailbox with specified, potentially dangerous attachments such as .vbs, .exe, bat, and .pif.

There are many anti-virus (free as well as commercial) suites or walls available that can integrate with Q-Mail without compromising on any feature. Some of these are:

- Q-Mail-Scanner—Open source and free
- Trend’s InterScan VirusWall Virus scanner
- Sophos’ ‘sweep’ virus scanner

- H + BEDV’s antivirus scanner
- Kaspersky’s AVLinux scanner
- McAfee’s (NAI’s) virus scanner
- Command’s virus scanner
- F-Secure anti-virus scanner
- F-Prot anti-virus scanner
- Inoculan znti-virus scanner
- Clam Antivirus—an Open Source antivirus scanner
- RAV Antivirus

In our implementation, we have chosen Trend Micro’s InterScan VirusWall. In spite of the fact that InterScan VirusWall was built for Sendmail, a smooth integration was done with Q-Mail.

```

# tar xvf isvwnux.tar
# ./linux/isinst

```

will give a text-based menu and is easily understandable. After installation, activate the Web browser to http://ip\_address\_of\_host:1812/interScan and you can configure it. Now comes the tricky part—for Sendmail, the defaults will be fine but will not work for Q-Mail properly. Changes that need to be done are (via Web interface)

1. Configuration—> E-Mail Scan Configuration—> Choose Daemon port instead of local server and enter a port number, say 825.
2. Save the settings
3. Add a line in /etc/services as  
qmail-smtp 825/tcp
4. Replace smtp by qmail-smtp in file /service/qmail-smtpd/run
5. Turn ON the Mail service of InterScan via the Web interface

You can now test it by

```

#telnet 0 25
You should get output as
Trying 0.0.0.0...
Connected to 0 (0.0.0.0).
Escape character is '^]'.
220-InterScan Version 3.6-Build_1166 $Date: 04/24/2001
22:13:0052$: Ready
220 mail.linuxforyou.com ESMTP

```

Now you are ready to use the VirusWall activated Mailserver.

The best way to implement in such environments is to go through the article and keep implementing and testing. While going through this implementation, you might feel that the setup is complex and difficult to implement. Here, the entire implementation did not take longer than a few hours. This article is only the first step towards enterprise messaging solutions, which have a great future ahead. **LFY**

*The article is written by Mr Biswajit Banerjee, Director, Tetra Information Services Pvt. Ltd, who are into Linux corporate solutions. He can be contacted at biswajit@tetra.in.com*