



welcome to

## The qmailrocks.org qmail Installation guide for FreeBSD

A word on the FreeBSD Qmail installation:







This qmail installation guide for FreeBSD has been successfully tested on FreeBSD 4.x and 5.x with no problems. I have not tested this installation version previous to 4.x. However, I'm relatively sure that this installation could be applied to previous versions with minimal fuss. Secondly, when it comes to Qmail and FreeBSD there are basically 2 routes you can go

**(1) Installation from the FreeBSD ports collection** - Obviously this is the easier route to go. It's so easy that I'm not going to cover it. However, the FreeBSD ports installation of qmail is typically NOT a proper installation. It often places the files and folders for qmail in locations that Dan Bernstein himself never recommended or even suggested. For that reason, I wouldn't install it from the ports collection. Of course, that decision is up to you. But if you decide to install from the ports collection, don't come asking me for help. You're on your own, my friend.

**(2) Installation from source** - This is the way my installation guide does it. Why? Because it's predictable and you always know what you're getting and where everything is going. This makes troubleshooting and support much easier. Also, installing from source is far more platform independent than installing from the ports collection. This means that your qmail installation will be much more portable should you ever want to upgrade your system, change operating systems or even if you have to do a restoration after some sort of hardware failure. Sure it takes a bit longer, but you'll end up with a correctly installed, widely supported and easily portable implementation of qmail. Trust me, it's worth the extra time. Ok, enough of that. Let's get to it.

[Take me to the qmailrocks.org FreeBSD qmail installation guide!](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
	<b>Bold Black Text</b>   Commands to be run by you, the installer.
	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
	<b>Bold Red Text</b>   Vital and/or critical information.
	<b>Regular/Bold Purple text</b>   Denotes helpful tips and hints or hyperlinks.
	<b>Regular Orange Text</b>   Command line output.
	<b>Regular green text</b>   Denotes the contents of a file or script.





welcome to

## The qmailrocks.org qmail Installation guide for FreeBSD

As it is now, I've split the FreeBSD qmail installation up into 20 sections which are listed below. Simply start at step 1 and follow it all the way to completion. By the time you reach step 20, you should have a smokin' qmail installation of qmail blazing away on your server.

For a quick look at what this installation will provide you with, [click here](#).

**Before You Start!** - When installing qmail, I would STRONGLY recommend that you first conduct a test installation on a test server if you have one available. I cannot stress this enough. If you're new to qmail, chances are you will probably screw the install up the first time through. Do yourself a favor and screw up on a test server that has no importance to you! I had to learn this the hard way. ;)

Part 1 - [Download All the Needed Items for the Qmail installation](#)

Part 2 - [Installing qmail itself](#)

Part 3 - [Installing EZmlm and EZmlm-idx](#)

Part 4 - [Installing Qmail-Autoresponder](#)

Part 5 - [Installing Vpopmail](#)

Part 6 - [Installing VQadmin](#)

Part 7 - [Installing maildrop](#)

Part 8 - [Installing QmailAdmin](#)

Part 9 - [Finalizing the qmail installation](#)

Part 10 - [Uninstalling Sendmail](#)

Part 11 - [Starting up qmail](#)

Part 12 - [Installing Courier IMAP & IMAP SSL](#)

Part 13 - [Installation of the Squirrelmail web mail program](#)

Part 14 - [Installing Clam Antivirus & SpamAssassin](#)

Part 15 - [Installing qmail-scanner w/qms-analog](#)

Part 16 - [Installing Qmailanalog w/qlogtools & qms-analog](#)







Part 17 - [Installing Qtrap](#)

Part 18 - [Maintaining your qmail server](#)

Part 19 - [Mail client configuration](#)

Part 20 - [Feedback](#)

qmlrocks@earthlink.net

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
	<b>Black Text</b>	Commands to be run by you, the installer.
	<b>Maroon Text</b>	Special notes for Redhat 9 users.
	<b>Red Text</b>	Vital and/or critical information.
	<b>Purple text</b>	Denotes helpful tips and hints or hyperlinks.
	<b>Orange Text</b>	Command line output.
	<b>Green text</b>	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 1 - Download the Software

The first thing you'll need to do to get started is to download all the needed software packages for the entire qmail installation process. To make things easier for everyone, I've combined all the needed packages into 1 giant tarball (.tar.gz) bundle that you can download in one easy step. All of the included packages are the latest versions as of **November, 2004** (there are a few intentional exceptions) and I will make efforts keep the software bundle up to date as new versions are released. If your going to be using the Qmailrocks installation guide to install qmail, I would strongly recommend that you download my software bundle. I've included several ready-made scripts and so forth, so everything will make more sense if you're on the same page as I am when going through the install.

So let's start the installation by getting the needed software. You will notice that below I create a new directory called /downloads and I place the Qmailrocks tarball in that directory before unpacking it. I would strongly recommend you do this as the rest of the instructions on this site are geared toward this source directory structure. Anyway, lets get down to business.

**mkdir /downloads**

**cd /downloads**

Now download the Qmailrocks.org software bundle. The command below will download the bundle from the qmailrocks.org main server in Texas, USA. You can also download the bundle from a [qmailrocks mirror site](#), if you wish)

**wget http://www.qmailrocks.org/downloads/qmailrocks.tar.gz**

(Alternatively, if you'd like to download individual packages or view a listing of all the packages, you may do so [right here](#).)

Once you've downloaded qmailrocks.tar.gz, were going to place in a directory called "downloads" and then unpack it...

**tar zxvf qmailrocks.tar.gz**

[Proceed to Part 2](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>   Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>   Vital and/or critical information.
■	Regular/Bold Purple text   Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text   Command line output.
■	Regular green text   Denotes the contents of a file or script.



## Part 2 - Installing Qmail Itself

Now that you've downloaded all the needed packages, we can start the install. At this point you should have a qmailrocks source directory located at /downloads/qmailrocks. If you don't, go back to step 1.

To start things off, I've created a handy little shell script that takes care of the first portion of getting qmail, ucspi-tcp and daemontools intalled. Simply run this script from the command prompt of your Solaris box and you should be golden. The script will tell you what it's doing along the way.

**/downloads/qmailrocks/scripts/install/qmr\_install\_freebsd-s1.script** ([click here](#) to view this script)

If all goes well, you should have all the needed user and groups created as well as all the needed directories, permissions and ownership settings needed for the installation of qmail, ucspi-tcp and daemontools

Lastly, we apply 3 patches to qmail before we compile it. I've included a handy little script to apply all 3 patches at once. The 3 patches we are going to apply are: 1) The standard qmail-1.03 patch 2) The qmailqueue patch to enable the use of qmail--scanner later in this installation and 3) The auth-jms1.4a patch to enable SMTP authentication. Please note that if you downloaded the qmailrocks.tar.gz package before February 15th, 2004, you will need to download it again as the script below is a new feature in the qmailrocks package as of this date.

So let's patch qmail...

**/downloads/qmailrocks/scripts/util/qmail\_patches.script** ([click here to view this script](#))

Now we build Qmail...

```
cd /usr/src/qmail/qmail-1.03
```

```
make man
```

```
make setup check
```

```
./config-fast your_fqdn_hostname (ex: ./config-fast mail.mydomain.com)
```

If there are no errors, Qmail has been built successfully!

Now we build ucspi-tcp...

```
cd /usr/src/qmail/ucspi-tcp-0.88/
```

```
make
```

```
make setup check
```

If you don't get any errors, that's it for ucspi-tcp!

Now we build the daemontools....

```
cd /package/admin/daemontools-0.76
```

```
package/install
```

If no errors are reported, you've successfully compiled the daemontools package!

**Note: You may notice that after you install daemontools, the install script will tell you to reboot your server in order to start the svscan service. Take this advise and reboot your server now.**

**reboot**

When your server comes back online, a "ps -aux" command should reveal that the daemontools "svscan" service is now running. Woohoo!

OK, Qmail is almost totally installed but we're going to pause right here and install a bunch of handy tools and features that will make Qmail pretty and fun! After that, we'll make some final changes to Qmail and then crank it up!

[Proceed to Part 3](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>	Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 3 - EZmlm

EZmlm is a nice mailing list add-on to Qmail. I've used it several times myself and its actually one of the better mailing list programs out there. When we install Qmailadmin later on, you'll see that EZmlm integrates seamlessly into Qmailadmin to provide a very user friendly mailing list management interface. As an added bonus, Vpopmail (which we will install as well) will let you control what users can and cannot use mailing lists! Can't beat that!

So let's install it...

```
cd /downloads/qmailrocks/
```

```
tar zxvf ezmlm-0.53-idx-0.41.tar.gz
```

```
cd ezmlm-0.53-idx-0.41
```

```
make
```

```
make setup
```

If you don't get any errors, then ezmlm is all set up and ready to go!

[Proceed to Part 4](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>	Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.



## Part 4 - Qmail-Autoresponder

Qmail-Autoresponder does exactly what you think it does. It allows us to set up autoresponders for mailboxes and so forth. I've pre-compiled a FreeBSD package that should make the install a breeze. If you don't want to use the package I've got, you can always install Qmail-autoresponder from the FreeBSD ports collections. However, my package was made directly from the ports collection, so there should be no difference.

So let's install it...

The easiest way to install qmail-autoresponder is to use the FreeBSD package management tool...

**pkg\_add -r qmail-autoresponder**

This should fetch and install qmail-autoresponder in a jiffy. However, if you are unable to fetch the package, you can install it with the following alternative method...

**cd /downloads/qmailrocks**

**pkg\_add qmail-autoresponder-0.96.1.tbz**

Wow, that was easy. If you don't get any errors, then qmail-autoresponder is all set up and ready to go!

[Proceed to Part 5](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text Qmail installation notes and summaries by the author. Me talking.
<b>Black</b>	<b>Bold Black Text</b> Commands to be run by you, the installer.
<b>Maroon</b>	<b>Bold Maroon Text</b> Special notes for Redhat 9 users.
<b>Red</b>	<b>Bold Red Text</b> Vital and/or critical information.
<b>Purple</b>	<b>Regular/Bold Purple text</b> Denotes helpful tips and hints or hyperlinks.
<b>Orange</b>	<b>Regular Orange Text</b> Command line output.
<b>Green</b>	<b>Regular green text</b> Denotes the contents of a file or script.



## Part 5 - Vpopmail

### [I would like to install vpopmail without MySQL intregation](#)

(recommended for smaller email servers and for qmail/mysql newbies)

or

### [I would like to install vpopmail with MySQL integration](#)

(Requires that mysql server be installed on your server. Recommended for larger email servers & ONLY for experienced mysql users)

#### Which option is best for me?

First of all, don't feel that a vpopmail installation without mysql is somehow inferior or inadequate. It's not. The choice of whether or not to use mysql with vpopmail, in my opinion, is a personal preference and basically comes down to 2 things:

#### 1. How large is you mail server going to be?

If you are only planning on hosting a handful of domains on your mail server, I don't think it's really worth integrating mysql into it. I have a server that hosts about 50 domains on it right now and it does NOT have mysql integrated into vpopmail. It works perfectly fine. The qmailrocks.org mail server also does NOT have mysql built into it. It runs great. However, if you plan to host more than 50 domains or so, I'd say go with mysql. It makes it easier to manage a lot of domains and also makes porting the mail server to new equipment and locations easier. Of course, the decision is up to you. If you really want to use vpopmail with mysql on a server that hosts only 1 domain, knock yourself out. My opinion though, is that unless you are hosting a ton of domains, integrating mysql into vpopmail is simply making your mail server more complex than it needs to be. And as we all know, the more complex the plumbing, the easier it is to stop up the drain.

#### 2. How comfortable are you with mysql?

If you're a newbie and you don't know jack shit about mysql, don't be a jackass. Just use the default vpopmail installation and save yourself the aggravation. I can't hold you hand though installing and setting up mysql, and you'll be hard pressed to find someone else that will. In short, don't get in over your head.

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
<b>Black</b>	<b>Bold Black Text</b>   Commands to be run by you, the installer.
<b>Maroon</b>	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
<b>Red</b>	<b>Bold Red Text</b>   Vital and/or critical information.
<b>Purple</b>	<b>Regular/Bold Purple text</b>   Denotes helpful tips and hints or hyperlinks.
<b>Orange</b>	<b>Regular Orange Text</b>   Command line output.
<b>Green</b>	<b>Regular green text</b>   Denotes the contents of a file or script.



## Part 5 - Vpopmail

Vpopmail is one of the major components of this installation. Vpopmail allows us to do virtual domain mail hosting. It's got a lot of built in tools and features that make it a dream to work with. Even if you don't want to host mail for multiple domains, I would still recommend installing Vpopmail. It just makes the whole mail game easier. Plus, my installation centers around it, so if you don't install it you're going to have a headache.

So let's install it...

```
cd /downloads/qmailrocks
```

```
tar zxvf vpopmail-5.4.7.tar.gz
```

```
cd vpopmail-5.4.7
```

### - Helpful Hint: Vpopmail configuration options -

The vpopmail "configure" command can have loads of options. Use "./configure --help" to see them all. In the syntax used in this installation, I specify to the type of logging that I want vpopmail to use. Vpopmail logs its activities to the server's syslog and there are several options you can use. I've used the "p" option, but feel free to adjust it to your needs. Here's are the details:

```
--enable-logging=n - logs nothing
```

```
--enable-logging=e - logs only errors (default)
```

```
--enable-logging=y - logs all attempts
```

```
--enable-logging=p - logs errors with passwords
```

```
--enable-logging=v - verbose. Logs all attempts with passwords
```

Now let's configure vpopmail...







```
./configure --enable-logging=p
```

```
make && make install-strip
```

If you don't get any errors, then Vpopmail is good to go!

[Proceed to Part 6](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
	<b>Black Text</b>   Commands to be run by you, the installer.
	<b>Maroon Text</b>   Special notes for Redhat 9 users.
	<b>Red Text</b>   Vital and/or critical information.
	<b>Regular/Bold Purple text</b>   Denotes helpful tips and hints or hyperlinks.
	<b>Regular Orange Text</b>   Command line output.
	<b>Regular green text</b>   Denotes the contents of a file or script.

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 6 - Vqadmin

Vqadmin is simply a nice web based interface that will let us manage Vpopmail. Through the interface we can create new domains, new users, net quotas, enable services and much more.

So let's install it...

```
cd /downloads/qmailrocks
```

```
tar zxvf vqadmin-2.3.6.tar.gz
```

```
cd vqadmin-2.3.6
```

```
./configure --enable-cgibindir=/path/to/your/cgi-bin --enable-htmldir=/path/to/your/html/directory
```

(Example: ./configure --enable-cgibindir=/usr/local/www/cgi-bin --enable-htmldir=/usr/local/www/html )

```
make && make install-strip
```

If the installation is successful, Vqadmin should install itself in the cgi-bin directory of your default website. Unless you tell it otherwise, that usually defaults to /var/www/cgi-bin. You can specify another location in the ./configure command above.

Now you will need to add the following to your server's Apache configuration file (usually called httpd.conf)

```
<Directory "/path/to/your/cgi-bin/vqadmin">
deny from all
Options ExecCGI
AllowOverride AuthConfig
Order deny,allow
</Directory>
```

```
cd /path/to/your/cgi-bin/vqadmin
```

Now you will want to create a .htaccess file to password protect the Vqadmin interface. There should already be a .htaccess file in the vqadmin directory, so all you need to do is configure it.

```
vi .htaccess
```

```
AuthType Basic
AuthUserFile /path/to/where/you/want/to/store/the/password/file/.htpasswd
AuthName vQadmin
require valid-user
satisfy any
```

**chown www .htaccess** (you may need to change the chown to either "nobody", "apache" or "www" etc., depending on what user your installation of Apache is running as)

```
chmod 644 .htaccess
```

Now you need to create a corresponding .htpasswd file that's going to contain the username and encrypted password for the Vqadmin administrator...

```
htpasswd -bc /path/to/where/you/want/to/store/the/password/file/.htpasswd admin admin_password
```

```
chmod 644 /path/to/where/you/want/to/store/the/password/file/.htpasswd
```

**Question:** But what if I want to user another username other than "admin"?

**Answer::** You will notice that in the above line, I'm adding an admin user called "admin". The name of the user needs to be "admin" because that is the username which Vqadmin sets up by default to have full admin rights. If you want to use a username other than "admin", you will need to edit the /cgi-bin/vqadmin/vqadmin.acl file and add your custom user to that file along with whatever rights you want it to have. Within that file, you will see where the user called "admin" is already set up to have all rights. That line looks like this:

```
admin VIMUDCA admin1user
```

In this line, the "admin" part specifies the username, the "VIMUDCA" part specifies that user's rights (a chart of all the possible rights is right above this line), and the "admin1user" part specifies the common name for the user which will be displayed when you log into Vqadmin. The common name is purely for aesthetic purposes.

Now restart Apache...

```
apachectl stop
```

```
apachectl start
```

If all has gone well, you should now be able to browse (in your web browser) to:

<http://www.yourdomain.com/cgi-bin/vqadmin/vqadmin.cgi>







**Note:** make sure you include "vqadmin.cgi" in the path or you will get a 403 forbidden error!

Enter the user "admin" and whatever password your assigned it.

You should now see the Vqadmin interface. Go ahead and add a new domain to your server! Pretty cool, huh? The "postmaster" user serves as the admin user for any new domain and we will use it to log into Qmailadmin, when we install that. As soon as we complete the install, that domain will be instantly able to get mail (assuming the MX is correctly pointing to your server). The nice thing about Vpopmail and Vqadmin is that you do not need to restart anything after you add a domain. Once you add it, it simply works! We're almost done!

[Proceed to Part 7](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
	<b>Bold Black Text</b>   Commands to be run by you, the installer.
	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
	<b>Bold Red Text</b>   Vital and/or critical information.
	Regular/Bold Purple text   Denotes helpful tips and hints or hyperlinks.
	Regular Orange Text   Command line output.
	Regular green text   Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 7 - Maildrop

Maildrop is a mail filtering agent which can be used to filter messages as they arrive on the server.

Let's install it...

```
cd /downloads/qmailrocks
```

```
tar zxvf maildrop-1.6.3.tar.gz
```

```
cd maildrop-1.6.3
```

```
./configure --prefix=/usr/local --exec-prefix=/usr/local --enable-maildrop-uid=root --enable-maildrop-gid=vchkw --enable-maildirquota
```

```
make && make install-strip && make install-man
```

If you didn't get any errors, maildrop should be all set!

[Proceed to Part 8](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>   Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>   Vital and/or critical information.
■	Regular/Bold Purple text   Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text   Command line output.
■	Regular green text   Denotes the contents of a file or script.



## Part 8 - Qmailadmin

Qmailadmin is going to provide us with a nice web based interface for administering mail accounts once they are setup through Vpopmail (or Vqadmin). From Qmailadmin we can create mailboxes, aliases, forwards, mail robots, mailing lists. You'll also find a few other handy functions as well. Qmailadmin is sort of the icing on the Qmail cake.

Let's install it...



[click for full size preview](#)

```
cd /downloads/qmailrocks
```

```
tar zxvf qmailadmin-1.2.3.tar.gz
```

```
cd qmailadmin-1.2.3
```

```
./configure --enable-cgibindir=/path/to/your/cgi-bin --enable-htmldir=/path/to/your/html/directory --enable-autoresponder-path=/usr/local/bin/qmail-autoresponder
```

**note:** The paths in the above configure script will need to be custom tailored to your systems configuration

```
make && make install-strip
```

That's it! Now browse to <http://www.yourdomain.com/cgi-bin/qmailadmin> and you should see the login screen. Login with the postmaster account and password for the domain that you created a while back using Vqadmin. Pretty cool, isn't it? Go ahead and create some additional mailboxes for your domain(s).

If you didn't get any errors, Qmailadmin should be all set!

**note:** if it's late and you're looking for a place to stop so you can sleep, this is a good place. Sendmail should still be handling mail on the server. After this page it's Qmail or bust!

[Proceed to Part 9](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>   Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>   Vital and/or critical information.
■	Regular/Bold Purple text   Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text   Command line output.
■	Regular green text   Denotes the contents of a file or script.



## Part 9 - Finalizing Qmail

Ok, we've installed a bunch of bells of whistles onto our qmail installation. Now it's time to wrap up the configuration for qmail itself. After that, we will stop and remove Sendmail from the server and then it's time to crank qmail up!

The first thing we're going to do is create the qmail supervise scripts, create the the qmail rc and qmailctl scripts and then set the needed permissions on all these scripts. Lucky for you, I've created a script to do all this for you. The script will give you a breakdown of what it is doing while it's running. If any errors occur, you'll see them. However, if you've configured everything right up until now, you shouldn't have any problems. You can check out the contents of this scripts [right here](#).

So let's run the script...

```
/downloads/qmailrocks/scripts/finalize/freebsd/finalize_freebsd.script
```

Hey, that was easy. Now there are just a couple tweaks to make to these new scripts we just created...

```
vi /var/qmail/supervise/qmail-pop3d/run
```

Find "**mail.example.com**" and change it to your server's hostname. [For example: mail.mydomain.com.](#)

```
vi /var/qmail/supervise/qmail-smtpd/run
```

Find "**mail.example.com**" and change it to your server's hostname. [For example: mail.mydomain.com](#)

OK, all done there. By now, you may notice that some Qmail functions are already up and running, so to finish the install, we will stop Qmail....

```
qmailctl stop
```

And setup elective relaying...

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

```
qmailctl cdb
```

```
echo some_address > /var/qmail/alias/qmail-root
```

where "some\_address" is the system user or email address you want these addresses aliased to.

```
echo some_address > /var/qmail/alias/qmail-postmaster
```

where "some\_address" is the system user or email address you want these addresses aliased to.

```
echo some_address > /var/qmail/alias/qmail-mailer-daemon
```

where "some\_address" is the system user or email address you want these addresses aliased to.

```
ln -s /var/qmail/alias/qmail-root /var/qmail/alias/qmail-anonymous
```

```
chmod 644 /var/qmail/alias/qmail*
```

Alright. We've got qmail ready to go. One of the last things we need to do is to disable/uninstall Sendmail on the server and replace the Sendmail binary with a symlink to qmail, so that our server won't freak out with Sendmail being gone.

[Proceed to Part 10](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Black Text</b>	Commands to be run by you, the installer.
■	<b>Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 10 - Uninstalling Sendmail

Well, the moment you've been waiting for is finally here. We're going to uninstall Sendmail from the server. However, since Sendmail is such a commonly used item among tons of server operations and cronjobs, you will see that, after we uninstall Sendmail, we will actually make an artificial Sendmail that is nothing more than a direct injection into Qmail.

Anyway, let's uninstall it...

**killall sendmail** (to stop all sendmail processes)

**mv /usr/sbin/sendmail /usr/sbin/sendmail.old**

**mv /usr/lib/sendmail /usr/lib/sendmail.old** (this may not apply if there is no sendmail link or binary in this location)

**chmod 0 /usr/lib/sendmail.old /usr/sbin/sendmail.old**

Now we will need to instruct FreeBSD not to attempt to start Sendmail upon startup. This is done by make the following modification to the /etc/rc.conf file:

Change **sendmail\_enable="YES"** to **sendmail\_enable="NONE"**

Now save the file and exit out of it.

Finally, we will need to set up an "artificial" Sendmail, which is just a symbolic link to Qmail's Sendmail. This is needed to ensure that the myriad of systemic mail scripts are still able to send mail! Qmail's "Sendmail" is nothing more than a direct injection into Qmail itself...

**ln -s /var/qmail/bin/sendmail /usr/lib/sendmail**







**ln -s /var/qmail/bin/sendmail /usr/sbin/sendmail**

That's it! If all has gone well, Sendmail should be uninstalled and the Qmail Sendmail should be in its place.

Now we'll will finalize the installation of Qmail and then crank it up!

[Proceed to Part 11](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
	<b>Bold Black Text</b>	Commands to be run by you, the installer.
	<b>Bold Maroon Text</b>	Special notes for Redhat 9 users.
	<b>Bold Red Text</b>	Vital and/or critical information.
	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
	Regular Orange Text	Command line output.
	Regular green text	Denotes the contents of a file or script.



## Part 11 - Starting up qmail

Alright, qmail should be ready to go! But before we crank it up, let's run a script that will check the key components of the installation and make sure everything is alright.

To do this test, I have borrowed Dave Sill's "inst\_check" script, but I've made a few custom modifications to accomodate for the subtle differences between the Qmailrocks installation and the Life With Qmail installation. Basically, the Qmailrocks installation has a slightly different logging setup and some slight variations in permissions settings. If you've installed according to this site, use my version of the script, as using Dave's version will result in a lot of "error" detections that are false positives due to the differences in the 2 installs. When you run the script, it will check for some key required files and folders and will also check permissions and owership settings on many key items. It a needed file does not exist or if the ownership/permissions settings are wrong on a key file, it will tell you and then make a suggestion as to how to correct the error. **This script does NOT check the CONTENT or SYNTAX of your scripts, but only for the scripts' existence and their ownership/permissions settings. If you've screwed up the syntax of on the run scripts, this tool will not detect it.** So you ready? Let's do it...

**/downloads/qmailrocks/scripts/util/qmr\_inst\_check**

If you get a "congratulations" type of message, you're all set. If you get some errors, just follow the directions to fix the errors and then re-run the script until you get all errors corrected and you get a "congratulations" message.

Assuming, you've passed the installation check script, let's crank Qmail up!

**qmailctl stop**

**qmailctl start**

You can find out how things are running by:

**qmailctl stat**

You should see an output like this:

```

/service/qmail-send: up (pid 29956) 2 seconds
/service/qmail-send/log: up (pid 29960) 2 seconds
/service/qmail-smtpd: up (pid 29963) 2 seconds
/service/qmail-smtpd/log: up (pid 29968) 2 seconds
/service/qmail-pop3d: up (pid 29971) 2 seconds
/service/qmail-pop3d/log: up (pid 29972) 2 seconds
messages in queue: 0
messages in queue but not yet preprocessed: 0

```

If you, don't see anything like that or if you see error messages, [click here](#) for troubleshooting tips.

Congratulations, Qmail is now officially up and running and you should be able to send and receive mail on the server. You can test it with the domain you created in Vqadmin a while back.

**telnet localhost 110**

you should see something like this:

```

Trying 192.168.1.10...
Connected to 192.168.1.10.
Escape character is '^]'.
+OK <16658.1054485137@yourserver.com>
user postmaster@mydomain.com (enter your username here. remember to use the full e-mail address)
+OK
pass your_password

```

```
+OK
quit
+OK
Connection closed by foreign host.
```

This is the sign of a successful POP connection to the server!

Now try sending mail to that same user from another location. Telnet to 110 again and run the "list" command and you should see the message that you send...

**telnet localhost 110**

```
Trying 192.168.1.10...
Connected to 192.168.1.10.
Escape character is '^]'.
+OK <16658.1054485137@yourserver.comt>
user postmaster@mydomain.com (again, remember to log in with the full email address of the user)
+OK
pass your_password
+OK
list
+OK
1 323 (there's your message!)
.
quit
+OK
Connection closed by foreign host.
```

If you have reached this point, then Qmail is now successfully up and running. Technically speaking, you could quit right here and have a functioning mail server. However, we still have a few options that we're going to plug into Qmail before we're done. In the next pages we will:

Install Courier IMAP & IMAP SSL.

Install Squirrelmail web based mail interface (requires that IMAP be installed).

Install Qmail-Scanner, an alternative queuing device.

Install Spamassassin, to tag all incoming spam.

Install Clam Anti Virus - To quarantine e-mails containing known viruses.

Install Checkall, to filter out messages containing undesirable words that may make it past Spamassassin.

[Proceed to Part 12](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text Qmail installation notes and summaries by the author. Me talking.
<b>Black</b>	<b>Bold Black Text</b> Commands to be run by you, the installer.
<b>Maroon</b>	<b>Bold Maroon Text</b> Special notes for Redhat 9 users.
<b>Red</b>	<b>Bold Red Text</b> Vital and/or critical information.
<b>Purple</b>	<b>Regular/Bold Purple text</b> Denotes helpful tips and hints or hyperlinks.
<b>Orange</b>	<b>Regular Orange Text</b> Command line output.
<b>Green</b>	<b>Regular green text</b> Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 12 - Installing Courier IMAP & IMAP SSL

Now that you have Qmail up and running, we're going to add a few extras onto it. For starters, we're going to install Courier IMAP & IMAP SSL. This will, obviously, enable IMAP connections to the mail server and it is a necessary ingredient for most popular web based mail clients such as Horde, SQwebmail and Squirrelmail. Courier IMAP is the preferred IMAP server to install because it has built in support the vchkw mail user setup that Vpopmail utilizes. In short, Courier IMAP works with Vpopmail and virtual domains.

So let's install it.....

```
cd /downloads/qmailrocks/
```

```
bunzip courier-imap-3.0.8.tar.bz2 (Note: some systems may use bunzip2 instead of bunzip)
```

```
tar xvf courier-imap-3.0.8.tar
```

```
cd courier-imap-3.0.8
```

```
./configure --prefix=/usr/local --exec-prefix=/usr/local --with-authvchkw --without-authldap --disable-root-check --with-ssl
```

**Note: the configure process will take a few minutes. Go grab a snack...**

```
make && make install-strip
```

```
cd /usr/local/etc
```

Make sure that the files "imapd" and "imapd-ssl" exist. If they do not exist, do the following:

```
cp imapd.dist imapd
```

```
cp imapd-ssl.dist imapd-ssl
```

```
cp authdaemonrc.dist authdaemonrc
```

```
vi imapd-ssl
```

```
make sure IMAPDSSLSTART=YES
```

Save and exit.

### Special note for people running a small home or office network:

If you are planning on having multiple users connect to your IMAP server from a single IP address, such as in a small home or office network, you may want to increase the "MAXPERIP" setting with the `/usr/local/etc/imapd` config file. This setting establishes the maximum number of IMAP connections that can be made from a single IP address. An example of this might be if you have a small office network running on a single DSL or Cable IP address and your mail server is outside of that network. While each computer in your internal network may have it's own private IP address, to the outside world anyone coming from your network has the single routeable IP address assigned to your DSL or Cable connection. The default setting for "MAXPERIP" is 4 so if you have a similar network setup and more than 4 people trying to access your IMAP server, you may want to increase this setting accordingly to avoid connection errors. Within the `/usr/local/etc/imapd` file, the line you are looking for looks like this:

```
MAXPERIP=4
```

**cp /usr/local/libexec/imapd.rc /usr/local/etc/rc.d/imap.sh**

**cp /usr/local/libexec/imapd-ssl.rc /usr/local/etc/rc.d/imapd.sh**

Now let's start up IMAP and IMAP SSL...

**/usr/local/etc/rc.d/imap.sh start**

**/usr/local/etc/rc.d/imapd.sh start**

If IMAP is running correctly, a "ps -auxw" should show several processes labeled "/usr/local/libexec/authlib/authdaemon.plain start"

If you run "nmap localhost", you should see both 143 and 993 now open and listening.

Now let's test it...

**telnet localhost 143**

```
Trying 192.168.1.10...
Connected to 192.168.1.10.
Escape character is '^'.
* OK [CAPABILITY IMAP4rev1 UIDPLUS CHILDREN NAMESPACE THREAD=ORDEREDSUBJECT THREAD=REFERENCES SORT QUOTA IDLE
STARTTLS] Courier-IMAP ready. Copyright 1998-2003 Double Precision, Inc. See COPYING for distribution information.
a login postmaster@mydomain.com my_password
a OK LOGIN Ok. (successful login!)
a logout (logs you out)
* BYE Courier-IMAP server shutting down
a OK LOGOUT completed
Connection closed by foreign host.
```

**Hint: The "a" that you see before my login commands is required.**

If you were able to log in , as in the example above, you're all set. IMAP is installed!

[Proceed to Part 13](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
<b>■</b>	<b>Bold Black Text</b>   Commands to be run by you, the installer.
<b>■</b>	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
<b>■</b>	<b>Bold Red Text</b>   Vital and/or critical information.
<b>■</b>	Regular/Bold Purple text   Denotes helpful tips and hints or hyperlinks.
<b>■</b>	Regular Orange Text   Command line output.
<b>■</b>	Regular green text   Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 13 - Installing Squirrelmail

Now that we have Qmail running with IMAP, we can install a webmail client to make mail accessible via a web browser. My choice for this was [Squirrelmail](#). Squirrelmail is both easy to install and it has lots of nice plugins to broaden its abilities. I know that a lot of people out there like to use [Horde](#). I like Horde myself and I've installed it on my Qmail server alongside Squirrelmail. However, Horde is a major pain in the ass to install. Anyone who's ever intalled it will tell you that. I just don't want to take the responsibility for it on this site. Some people also like to use SQwebmail. No offense, but I don't like SQwebmail.

Anyway, let's install Squirrelmail...

The first order of business to make sure PHP is installed and correctly configured. So let's get that out of the way...

In order for Squirrelmail to work correctly, you'll need to check a couple things about your PHP installation:

1. First of all, make sure some rendition of PHP 4 is installed. If it's not, kick yourself in the ass and then go install it. Sorry, I'm not going to give a PHP installation tutorial. To be safe, you will want the following config options to be active in your PHP installation.

```
--enable-track-vars
--enable-force-cgi-redirect
--with-gettext
--with-mysql
```

2. Make sure you have PHP uploads turned ON. This is done by editing a line in your php.ini file. The location of the php.ini file can vary, but it's usually located at /usr/local/etc/php.ini. If it's not, don't panic. Just run the old "locate php.ini" command. ;) Here's the line you will want to check/edit:

```
file_uploads = On
```

That's it for the PHP setup. Now let's download Squirrelmail...

You can download the latest stable version of Squirrelmail from: <http://www.squirrelmail.org/download.php>

I recommend downloading the .tar.gz version of the latest release.

Now change directories to the web directory of the website you want to serve Squirrelmail off of. In my case, I used /var/www/html

```
tar zxvf /path/to/squirrelmail-x.x.x.tar.gz (enter whatever version you downloaded)
```

Now rename the untarred folder to something more friendly...

```
mv squirrelmail-x.x.x webmail
```

And now let's configure Squirrelmail...

```
mkdir /var/sqattachments
```

```
chown -R www:www /var/sqattachments (or whatever user apache runs as)
```

```
cd webmail
```

```
chown -R www:www data (or whatever user apache runs as)
```

```
cd config
```

```
./conf.pl
```

This will run the Squirrelmail setup script which will allow you to customize the installation as well as set your server settings. Most of the important settings are in area #2, which is dubbed "Server Settings". Here are the specs I recommend:

- General  
-----
1. Domain : 1.2.3.4 (**Enter the IP of your server here. Don't be an idiot and actually use 1.2.3.4**)
  2. Invert Time : false
  3. Sendmail or SMTP : SMTP

- IMAP Settings  
-----
4. IMAP Server : localhost
  5. IMAP Port : 143
  6. Authentication type : login
  7. Secure IMAP (TLS) : false
  8. Server software : other
  9. Delimiter : detect

- SMTP Settings  
-----
4. SMTP Server : localhost
  5. SMTP Port : 25
  6. POP before SMTP : false
  7. SMTP Authentication : login
  8. Secure SMTP (TLS) : false

Depending on what version of Squirrelmail you are installing, the setup menu may differ slightly. But you get the idea. If you like, there a several other features of Squirrelmail you can customize that, while not critical, are sometimes fun. Also, check out [Squirrelmail's site](#) for tons of cool plugins. Make sure you save all settings before exiting the configuration menu.

Once you've configured Squirrelmail to your liking, go test it!.....

<http://www.yourdomain.com/webmail>

We'll sign in with the postmaster account under the domain you should have created earlier using Vqadmin...

**Username: postmaster@yourdomain.com**

**Password: your\_password**

If all has gone well, Squirrelmail should log your right into your account! From here you will be able to both send and receive mail as well as a host of other additional functions. Again, Squirrelmail has tons of really cool plugins, and you can check them out at [Squirrelmail's plugins page](#). Installing the plugins is pretty easy and their site can help you out. Now that was nice and painless, wasn't it? If I had tried to explain installing Horde instead, you would probably be holding a gun to your head right now, wishing for quick end to the misery. OK, I'm only kidding. :)

[Proceed to Part 14](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b>	Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Bold Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 14 - Clam Antivirus & Spamassassin

In this next step we will install Clam Antivirus and SpamAssassin. ClamAV is a global email virus scanner that scans all incoming email and quarantines infected files. SpamAssassin is a spam filter that grades each incoming e-mail based on a ruleset and then labels e-mail as spam should the e-mail message exceed a certain score. Once we have both of these jewels installed, we will tie them into qmail via qmail-scanner, which gets installed on the next step.

Now let's install Clam Antivirus and SpamAssassin..

The first thing you must do is make sure that you've got a recent version of Perl installed on your box. Using the command "**perl -v**" will typically tell you what version is running. Spamassassin 3, at least when I installed it, required Perl 5.8. If you're running something older than 5.8, you will likely need to upgrade. I'm not going to go into a great deal of detail into how to upgrade Perl here, but I will tell you a little story that might help. My FreeBSD box was running a default 5.005 version of perl. Using the "**pkg\_delete**" command, I removed that version. I then installed the new Perl 5.8 from the FreeBSD ports collection. After that, I ran the following command to instruct the system to use the ports installation of Perl from now on:

### use.perl port

And viola! My box was now using the new 5.8 version of Perl. I hope you like my little story.

Once you've got the right version of Perl installed on your box, you will now want to verify that you have all the required Perl modules and packages that will enable Clam AV and Spamassassin to function correctly.

The following Perl Modules/packages are required. If you find that I'm missing one here, let me know.

**Digest::SHA1**  
**Digest::HMAC**  
**Net::DNS**  
**Time::HiRes**  
**HTML::Tagset**  
**HTML::Parser**  
**Pod::Usage**  
**Parse::Syslog**  
**Statistics::Distributions**

### - Helpful Hints -

#### Checking/Installing Perl Modules

#### [How do I know if my server has these perl modules?](#)

The following script will check your system for the above modules.

**You MUST run this script as a NON-ROOT user!!**

**/downloads/qmailrocks/scripts/util/check\_perlmods.script**

If you see an output similar to this: `/usr/local/lib/perl5/site_perl/5.6.1/mach/Time/HiRes.pm`, you're good to go! However, if you get a "not found" type response for any of the above modules, you will need to install them.

#### [Ok, so if I don't have the module\(s\), how do I down and install it\(them\)?](#)

There are 3 main ways you can do this:

**1. The old fashioned way (CPAN)** - Go to <http://www.cpan.org>, download the needed module, read the "INSTALL" file and away you go. I've also

included tarballs of these modules in the Qmailrocks package, so feel free to use them if you want.

**2. Using the pkg\_add utility** - This is the best method because the pkg\_add utility AUTOMATICALLY installs any dependencies that may be needed for these Perl modules to be installed. This saves lots of time and hassle.

**3. The easy way (The Ports Collection)** - The FreeBSD ports collection has made a lazy man out of me. They make all this crap so easy.

#### Installing directly with the pkg\_add utility

As root, run the following commands:

```
pkg_add -r p5-Time-HiRes
pkg_add -r p5-Net-DNS
pkg_add -r p5-Digest-SHA1
pkg_add -r p5-Digest-HMAC
pkg_add -r p5-HTML-Tagset
pkg_add -r p5-HTML-Parser
pkg_add -r p5-Parse-Syslog
pkg_add -r p5-PodParser
pkg_add -r p5-Statistics-Distributions
```

#### Installing from the ports collection:

On my server, I found the needed ports in the following locations:

```
Time::HiRes - /usr/ports/devel/p5-Time-HiRes
Net::DNS - /usr/ports/dns/p5-Net-DNS
Digest::SHA1 - /usr/ports/security/p5-Digest-SHA1
Digest::HMAC - /usr/ports/security/p5-Digest-HMAC
HTML-Tagset - /usr/ports/www/p5-HTML-Tagset
HTML::Parser - /usr/ports/www/p5-HTML-Parser
Pod::Usage - /usr/ports/textproc/p5-PodParser
Parse::Syslog - /usr/ports/textproc/p5-Parse-Syslog
Statistics::Distributions - /usr/ports/math/p5-Statistics-Distributions
```

Simply cd into each packages directory as shown above and run: **make install && make clean**

**Note:** If you find that the directory "/usr/ports" does not exist, it means you don't have the ports collection. You can get the latest FreeBSD ports tree at <ftp://ftp.freebsd.org/pub/FreeBSD/ports/ports/ports.tar.gz> (anonymous login)

Damn! Now wasn't that easy?

Ok, so at this point you should have all 4 of the above Perl modules installed on your system. So let's install Clam Anti Virus and then Spamassassin.

Installing Clam Anti Virus...

```
cd /downloads/qmailrocks/
```

```
tar zxvf clamav-x.xx.tar.gz
```

```
cd clamav-x.xx
```

```
pw useradd clamav -s /sbin/nologin -d /tmp
```

```
pw useradd qscand -s /sbin/nologin -d /tmp
```

```
./configure
```

```
make && make check && make install && make clean
```

Now we'll do a few configuration settings to the `/usr/local/etc/clamd.conf` file. This file won't get use much, since we aren't going be using the daemonized version of ClamAV, but it does get accessed occasionally by other processes.

**vi /usr/local/etc/clamd.conf**

"Example" - needs be commented out. If it is not, make sure that it **is** commented out (#)  
"LogFile" - should be uncommented and set to `/var/log/clamav/clamd.log`  
"LogTime" - should be uncommmented.  
"LogSyslog" - should be uncommented.  
"User" - should be uncommented and set to **qscand**  
"ScanMail" - should be uncommented.

**vi /usr/local/etc/rc.d/clamd.sh**

```
#!/bin/sh
```

```
FOO_BIN=/usr/local/sbin/clamd  
test -x $FOO_BIN || exit 5
```

```
case "$1" in  
start)  
echo "Starting ` $FOO_BIN -V`"  
$FOO_BIN  
  
;;  
stop)  
echo "Shutting down ` $FOO_BIN -V`"  
killall $FOO_BIN
```

```
;;  
restart)  
$0 stop  
$0 start
```

```
;;  
*)  
echo "Usage: $0 {start|stop|restart}"  
exit 1  
;;  
esac
```

**chmod 755 /usr/local/etc/rc.d/clamd.sh**

**/usr/local/etc/rc.d/clam.sh start**

Ok, Clam AV is now installed, but let's go ahead and set it up so that it will auto-update every night with the latest virus definitions! First we will want to set up the proper logging for the updater....

**/usr/local/bin/freshclam -l /var/log/clamav/clam-update.log**

If the server is able to get updates, you should see an output similar to this:

```
[root@crescent clamav]# freshclam -l /var/log/clamav/clamav-update.log  
ClamAV update process started at Sun Oct 24 23:36:22 2004  
main.cvd is up to date (version: 27, sigs: 23982, f-level: 2, builder: tomek)  
daily.cvd is up to date (version: 549, sigs: 1583, f-level: 3, builder: ccordes)
```

Woohoo! You're updated with the latest virus definations from the Clam database!

Now we just set a crontab to run every night, which will run the auto-update procedure! In the example below, I've set mine to run every day at 1:40 AM. The odd run time came reccomended from ClamAV, if you're wondering. Running freshclam at times other than the top of an hour reduces the load on their servers. so feel free to adjust the time to something even more offbeat, like 1:42 AM or something. The guys at ClamAV will thank you.

**crontab -e** (make sure you run this command as root)

```
40 1* * * /usr/local/bin/freshclam --quiet -l /var/log/clamav/clam-update.log
```

Now save your new crontab and exit. That's it! We're all done with Clam Anti Virus! You will now have a server-wide e-mail virus scanner that updates itself every night with the latest and greatest virus definitions!

Now we install SpamAssassin...

**cd /downloads/qmailrocks**

Install most current version of SpamAssassin using the pkg\_add utility...

```
pkg_add -r Mail-SpamAssassin
```

If you are unable to getch SpamAssassin in the above command, I've included a copy of the package in the qmailrocks bundle. I'm so nice.

```
pkg_add p5-Mail-SpamAssassin-3.0.1_2.tgz
```

**Note:** Alternatively, if you're a source junkie, I've included a tarball of Spamassassin 3.01 in the qmailrocks bundle.

SpamAssassin is now installed, so let's configure it...

```
cd /usr/local/etc/mail/spamassassin
```

```
cp local.cf.sample local.cf
```

```
vi local.cf
```

Make sure the following lines exist and are not commented out.

```
rewrite_subject 1  
required_hits 5
```

Now save and exit out of the file.

And now we will configure the server to start SpamAssassin on boot:

```
vi /etc/rc.conf
```

Add the following line:

```
spamd_enable="YES"
```

Now save and exit out of the file.

Start up SpamAssassin:

```
/usr/local/etc/rc.d/sa-spamd.sh start
```

and check the status to make sure it's running:

```
/usr/local/etc/rc.d/sa-spamd.sh status
```

If spamd is running correctly, you should get an status output like this:







```
spamd is running as pid 65467 65470 65471 65472 65473 65474.
```

If all has gone well, both Spamassassin and Clam Anti Virus should now be installed and SpamAssassin should be running. With both of these

programs installed, we can now install Qmail-Scanner.

[Proceed to Part 15](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
	<b>Black Text</b>	Commands to be run by you, the installer.
	<b>Maroon Text</b>	Special notes for Redhat 9 users.
	<b>Red Text</b>	Vital and/or critical information.
	<b>Purple text</b>	Denotes helpful tips and hints or hyperlinks.
	<b>Orange Text</b>	Command line output.
	<b>Green text</b>	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 15 - qmail-scanner w/qms-analog

If you will recall, when we compiled qmail earlier in this installation, we applied a patch to qmail called "qmailqueue.patch". This patch allows qmail to be configured to run with a substitute queuing mechanism. That's exactly what we're about to do here. We're going to tell qmail to use Qmail-Scanner as the queuing mechanism. Qmail-scanner is going to allow us to integrate Clam Antivirus and SpamAssassin into our qmail server's mail queue. Once qmail-scanner is installed, there will be a master script that is filled with configuration options that help you to tailor the functionality of Clam Antivirus and SpamAssassin to your needs. To expand the number of configuration options, we are also going to apply a patch to qmail-scanner. For this patch, we will be using [Mark Teel's qms-analog patch](#). Qms-analog incorporated the widely used qmail-scanner-st patch but it also adds some cool reporting functionality as well which we will utilize later in this installation guide. So let's get on it!

**cd /downloads/qmailrocks**

Unpack qmail-scanner...

**tar zxvf qmail-scanner-1.24.tgz**

Now unpack qms-analog...

**tar zxvf qms-analog-0.4.0.tar.gz**

Install qms-analog itself. This will come in handy in the next step when we install Qmailanalog.

**cd qms-analog-0.4.0**

**gmake all**

Next, we copy needed qms-analog files to the qmail-scanner source directory...

**cp qmail-scanner-1.24-st-qms-*YYYYMMDD*.patch /downloads/qmailrocks/qmail-scanner-1.24/**

Now, let's apply the qms-analog patch...

**cd /downloads/qmailrocks/qmail-scanner-1.24**

**patch -p1 < qmail-scanner-1.24-st-qms-*YYYYMMDD*.patch**

Now we will configure qmail-scanner and install it. Ordinarily, you would run the ./configure script to configure and install qmail-scanner. However, Mark Teel has donated a handy little config script that does most of the work for you. This script is called "qms-config-script" and, if you look above, you should have already copied this config script into the qmail-scanner source directory.

How you go about configuring and installing qmail-scanner from this point on depends on how your server's installation of Perl is configured. For the purposes of this installation, there are 2 Perl setups.

1. Perl is configured to allow for setuid functions.
2. Perl is not configured for setuid functionality and, in fact, does not permit it.

We'll start off with the configuration step for a server that allows setuid. However, if you run into setuid errors, you can jump to a set of instructions for servers that do not allow setuid functionality.

So let's do it...

First, you need to configure the script for your needs...

```
cd /downloads/qmailrocks/qmail-scanner-1.24
```

```
vi qms-config
```

You will notice several fields that need to be customized to fit your needs. Let's have a look. I've highlighted the fields you should customize in **RED**

```
#!/bin/sh

if [ "$1" != "install" ]; then
INSTALL=
else
INSTALL="--install"
fi

./configure --domain yourdomain.com \
--admin postmaster \
--local-domains "yourdomain.com,yourotherdomain.com" \
--add-dscr-hdrs yes \
--dscr-hdrs-text "X-Antivirus-MYDOMAIN" \
--ignore-eol-check yes \
--sa-quarantine 0 \
--sa-delete 0 \
--sa-reject no \
--sa-subject ":SPAM:" \
--sa-delta 0 \
--sa-alt yes \
--sa-debug no \
--notify admin \
"$INSTALL"
```

Now save and exit out of the config file. That was easy, wasn't it.

And now we will run a test config for qmail-scanner...

```
chmod 755 qms-config
```

```
./qms-config
```

Answer YES to all questions. If you get no errors, you can then run the script in "install" mode and this will install qmail-scanner on your server.

**If the config test produced any error messages, stop here!**

If you got a "setuid" related error when you ran the above script it is most likely due to you Perl installation not supporting setuid functionality. You can [click here for alternate non-setuid configuration instructions](#).

If you get any other errors, check out these [troubleshooting tips](#).

If you didn't get any errors on the test run above, then you should be ok to run the "real" installation script below. So let's do it...

```
./qms-config install
```

Again, answer YES to all questions. If you get no errors, you can then run the script in "install" mode and this will install qmail-scanner on your server. If you do get errors, check out these [troubleshooting tips](#).

And now all that's left for qmail-scanner is to initiate the version file and the perlscanner database...

First, we'll initialize the version file. This command also helps to keep your server's /var/spool/qmailscan folder clear of rogue files that can develop when SMTP sessions are dropped. You may want to stick this command into your server's crontab and run it once a day. You'll see more on this in the "maintaining your qmail server" step near the end of this tutorial.. So let's run it...

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -z
```

And now we will generate a new perlscanner database for qmailp-scanner. For future reference, it's a good idea to run this next command whenever you upgrade qmail-scanner. You'll see more on this in the "maintaining your qmail server" step near the end of this tutorial. So let's do it...

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -g
```

A successful database build should produce the following output:

```
perlscanner: generate new DB file from /var/spool/qmailscan/quarantine-attachments.txt
perlscanner: total of 9 entries.
```

And now one final ownership check...

```
chown -R qscand:qscand /var/spool/qmailscan
```

Woohoo, qmail-scanner is installed! Now it's time to tie qmail-scanner into qmail itself.

```
vi /var/qmail/supervise/qmail-smtpd/run
```

To instruct Qmail to use Qmail-Scanner as the alternative queuing mechanism, we add the following line to the SMTP "run" script right under the first line (#!/bin/sh):

```
QMAILQUEUE="/var/qmail/bin/qmail-scanner-queue.pl" export QMAILQUEUE
```

..and we change the "softlimit" in that same script...

```
change softlimit to 40000000
```

**Note: It is absolutely vital that you change the "Softlimit" setting in this script. If you don't, qmail may fail to deliver mail!!!**

So now the qmail-smtpd/run file should look like this:

```
#!/bin/sh
QMAILQUEUE="/var/qmail/bin/qmail-scanner-queue.pl" export QMAILQUEUE
QMAILDUID=`id -u vpopmail`
NOFILESGID=`id -g vpopmail`
MAXSMTPD=`cat /var/qmail/control/concurrencyincoming`
LOCAL=`head -1 /var/qmail/control/me`
if [ -z "$QMAILDUID" -o -z "$NOFILESGID" -o -z "$MAXSMTPD" -o -z "$LOCAL" ]; then
echo QMAILDUID, NOFILESGID, MAXSMTPD, or LOCAL is unset in
echo /var/qmail/supervise/qmail-smtpd/run
exit 1
fi
if [ ! -f /var/qmail/control/rcpthosts ]; then
echo "No /var/qmail/control/rcpthosts!"
echo "Refusing to start SMTP listener because it'll create an open relay"
exit 1
fi
exec /usr/local/bin/softlimit -m 40000000 \
/usr/local/bin/tcpserver -v -R -l "$LOCAL" -x /etc/tcp.smtp.cdb -c "$MAXSMTPD" \
-u "$QMAILDUID" -g "$NOFILESGID" 0 smtp \
/var/qmail/bin/qmail-smtpd your_domain.com \
/home/vpopmail/bin/vchkpw /usr/bin/true 2>&1
```

Once you've got the qmail-smtpd file modified, save the changes and exit from the file. Now we will finalize the qmail-scanner installation by going over some post-install configuration options. After that, we'll fire everything up and take qmail-scanner for a test drive!

[Click Here to continue](#)

**If your Perl installation does NOT allow for setuid functionality**

```
cd /downloads/qmailrocks/qmail-scanner-1.24/contrib
```

```
make install
```

Now we will customize the qmail-scanner configuration script...

```
cd /downloads/qmailrocks/qmail-scanner-1.24
```

```
vi qms-config-cwrapper
```

You will notice several fields that need to be customized to fit your needs. Let's have a look. I've highlighted the fields you should customize in **RED**

```
#!/bin/sh

if [ "$1" != "install" ]; then
INSTALL=
else
INSTALL="--install"
fi

./configure --domain yourdomain.com \
--admin postmaster \
--local-domains "yourdomain.com,yourotherdomain.com" \
--add-dscr-hdrs yes \
--dscr-hdrs-text "X-Antivirus-MYDOMAIN" \
--ignore-eol-check yes \
--sa-quarantine 0 \
--sa-delete 0 \
--sa-reject no \
--sa-subject ":SPAM:" \
--sa-delta 0 \
--sa-alt yes \
--sa-debug no \
--notify admin \
--skip-setuid-test \
"$INSTALL"
```

Now save and exit out of the config file. That was easy, wasn't it.

And now we will run a test config for qmail-scanner...

```
chmod 755 qms-config-cwrapper
```

```
./qms-config-cwrapper
```

Answer YES to all questions. If you get no errors, you can then run the script in "install" mode and this will install qmail-scanner on your server. If you do get errors, check out these [troubleshooting tips](#).

```
./qms-config-cwrapper install
```

Again, answer YES to all questions. If you get no errors, you can then run the script in "install" mode and this will install qmail-scanner on your server. If you do get errors, check out these [troubleshooting tips](#).

```
vi /var/qmail/bin/qmail-scanner-queue.pl
```

Then change the first line of /var/qmail/bin/qmail-scanner-queue.pl to `#!/usr/bin/perl` (in other words, remove the `-T` from the perl call.)

```
chmod 0755 /var/qmail/bin/qmail-scanner-queue.pl
```

And now all that's left for qmail-scanner is to initiate the version file and the perlscanner database...

First, we'll initialize the version file. This command also helps to keep your server's /var/spool/qmailscan folder clear of rogue files that can develop when SMTP sessions are dropped. You may want to stick this command into your server's crontab and run it once a day. You'll see more on this in the "maintaining your qmail server" step near the end of this tutorial.. So let's run it...

**/var/qmail/bin/qmail-scanner-queue -z**

And now we will generate a new perlscanner database for qmailp-scanner. For future reference, it's a good idea to run this next command whenever you upgrade qmail-scanner. You'll see more on this in the "maintaining your qmail server" step near the end of this tutorial. So let's do it...

**/var/qmail/bin/qmail-scanner-queue -g**

A successful database build should produce the following output:

```
perlscanner: generate new DB file from /var/spool/qmailscan/quarantine-attachments.txt
perlscanner: total of 9 entries.
```

And now one final ownership check...

**chown -R qscand:qscand /var/spool/qmailscan**

Woohoo, qmail-scanner is installed! Now it's time to tie qmail-scanner into qmail itself.

**vi /var/qmail/supervise/qmail-smtpd/run**

To instruct Qmail to use Qmail-Scanner as the alternative queuing mechanism, we add the following line to the SMTP "run" script right under the first line (#!/bin/sh):

**QMAILQUEUE="/var/qmail/bin/qmail-scanner-queue" export QMAILQUEUE**

..and we change the "softlimit" in that same script...

**change softlimit to 4000000**

**Note: It is absolutely vital that you change the "Softlimit" setting in this script. If you don't, qmail may fail to deliver mail!!!**

So now the qmail-smtpd/run file should look like this:

```
#!/bin/sh
QMAILQUEUE="/var/qmail/bin/qmail-scanner-queue" export QMAILQUEUE
QMAILDUID=`id -u vpopmail`
NOFILESGID=`id -g vpopmail`
MAXSMTPD=`cat /var/qmail/control/concurrencyincoming`
LOCAL=`head -1 /var/qmail/control/me`
if [ -z "$QMAILDUID" -o -z "$NOFILESGID" -o -z "$MAXSMTPD" -o -z "$LOCAL" ]; then
echo QMAILDUID, NOFILESGID, MAXSMTPD, or LOCAL is unset in
echo /var/qmail/supervise/qmail-smtpd/run
exit 1
fi
if [ ! -f /var/qmail/control/rcpthosts ]; then
echo "No /var/qmail/control/rcpthosts!"
echo "Refusing to start SMTP listener because it'll create an open relay"
exit 1
fi
exec /usr/local/bin/softlimit -m 4000000 \
/usr/local/bin/tcpserver -v -R -l "$LOCAL" -x /etc/tcp.smtp.cdb -c "$MAXSMTPD" \
-u "$QMAILDUID" -g "$NOFILESGID" 0 smtp \
/var/qmail/bin/qmail-smtpd your_domain.com \
/home/vpopmail/bin/vchkpw /usr/bin/true 2>&1
```

Once you've got the qmail-smtpd file modified, save the changes and exit from the file. Now we will finalize the qmail-scanner installation by going over some post-install configuration options. After that, we'll fire everything up and take qmail-scanner for a test drive!

[Click Here to continue](#)

Let's go over some of the post-install configuration options for qmail-scanner...

### - Helpful Hints -

#### Post Install configuration tips for Qmail-Scanner

Although Qmail-Scanner should work pretty much "out of the box" so to speak, you can make some customizations to it's configuration by editing the qmail-scanner-queue.pl script located at `/var/qmail/bin/qmail-scanner-queue.pl`. The qmail-scanner-queue.pl script controls a lot of the functionality of both Clam AV and Spamassassin. Check it out for yourself and you will see that there are quite a few items you have control over. I wouldn't recommend touching most of them. In fact, the only setting that I changed in mine is in the Spamassassin section:

#### Can I have Spamassassin tag suspected spam with a custom subject line?

Yes. Edit the `/var/qmail/bin/qmail-scanner-queue.pl` file and find the following line:

```
my $spamc_subject='';
```

Now type a custom spam subject. This subject line will be added to any mails that Spamassassin tags as suspected spam. Here's an example:

```
my $spamc_subject='Hi, I'm Spam';
```

The "spamc\_subject" setting determines what message Spamassassin will append to the "subject" of e-mails which it deems as SPAM.

#### Can I delete e-mails that Spamassassin labels as spam?

Yes. Edit the `/var/qmail/bin/qmail-scanner-queue.pl` file and find the following line:

```
my $sa_delete='0';
```

Now replace the '0' with a number that represents how far above your SpamAssassin "required\_hits" variable that Qmail-scanner should start deleting messages at. For example, if you SpamAssassin required\_hits variable is set to "5" and you set the "sa\_delete" variable to "1.0", then any message that has a spam score of 1.0 over the "5" mark would be deleted. In other words, any mail with a score of 6 or more would be trashed automatically. So for this example, you would change the "sa\_delete" variable as follows:

```
my $sa_delete='1.0';
```

#### Is it safe to tell qmail-scanner to delete e-mails that SpamAssassin marks as spam?

Spamassassin has been tested to have up to a 99% accuracy rating in terms of detecting real spam and leaving legitimate e-mail alone. I've been using it for over a year now and have never gotten a false positive. Therefore, I feel safe in telling it to just delete the stuff.

There are a host of other Spam and Virus handling directives that can be customized with the qmail-scanner.pl file. You can check out the qmail-scanner patch website at <http://xoomer.virgilio.it/j.toribio/qmail-scanner/> for all the details.

Other than that, I left my qmail-scanner-queue.pl script as is.

To activate all the changes we just made, we're going to have to completely stop and restart Qmail. Since Qmail is so hard to kill (this is usually a good thing), I have found that the best thing to do at this point is to simply reboot the server. Rebooting will ensure that all processes are killed and restarted correctly. Now I'm sure I'll draw the typical Unix guru heat from this suggestion. Yes, I know that in theory you should never have to reboot a Unix server after it's first built. But hey, in my experiences with Qmail, a reboot usually is a good thing to get everything rolling together. If you don't believe me, try killing every Qmail process completely. You'll practically have to beat Qmail with a club to kill it. ;)

Testing the entire installation of Qmail-Scanner/Spamassassin/Clam AV:

Now it's time to test the whole damn thing to see if Qmail-Scanner, Spamassassin and Clam AV are all working correctly. Fortunately, Qmail-Scanner comes with it's own testing script that does a fantastic job. So let's test it!

```
cd /downloads/qmailrocks/qmail-scanner-1.24/contrib
```

```
chmod 755 test_installation.sh
```

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

A successful test should produce the following output and should send 4 new messages to whatever mailbox you specified in the Qmail-scanner configuration script above.

**setting QMAILQUEUE to /var/qmail/bin/qmail-scanner-queue.pl for this test...**

**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)...**

**Sending bad spam message for anti-spam testing - In case you are using SpamAssassin...  
Done!**

**Finished test. Now go and check Email for postmaster@mydomain.com**

When you check the mailbox to which the tests were sent, you should see 1 Qmail-scanner test message, 1 Spamassassin test message, 2 Clam AntiVirus virus detection messages. If the test script above does not produce any errors and if you get all 4 test messages in your administrative mailbox, you can pop open a beer and have a little celebration! You've successfully installed all 3 packages! Woohoo!

Summary of functionality:

If you've gotten to this point, you should have Clam Anti-Virus, Spamassassin and Qmail-Scanner all working together. When a messages comes into the server, Qmail-Scanner takes the message and pipes it out to both Clam Anti-Virus and Spamassassin. If the message contains a virus, Clam AV quarantines it a /var/spool/qmailscan/quarantine and then send a notification e-mail to whoever you specified in the Qmail-Scanner installation. If the message does not contain a virus, it is then scanned by Spamassassin. Depending on the score that Spamassassin assigns to the message and whether or not that score breaks the SPAM threshold set by you in the /var/qmail/.spamassassin/user\_prefs file, Spamassassin will either let the message go unaltered to its destination or it will tag the message as SPAM. If the message is tagged as SPAM, it will still arrive at its destination, but with an altered "subject" that will signal to the recipient that this was tagged as SPAM. The text that gets appended to the "subject" of the e-mail is set in the /var/qmail/bin/qmail-scanner.pl file. (For example: If you set qmail-scanner.pl to tag all SPAM with "HI, I'M SPAM!", mail tagged as such will be delivered to the recipient with "HI, I'M SPAM" added to the subject. Once the message is tagged, the recipient can then configure his/her mail client to deal with those tagged message in whatever manner he/she sees fit.

[Proceed to Part 16](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text Qmail installation notes and summaries by the author. Me talking.
<b>■</b>	<b>Bold Black Text</b> Commands to be run by you, the installer.
<b>■</b>	<b>Bold Maroon Text</b> Special notes for Redhat 9 users.
<b>■</b>	<b>Bold Red Text</b> Vital and/or critical information.
<b>■</b>	<b>Regular/Bold Purple text</b> Denotes helpful tips and hints or hyperlinks.
<b>■</b>	<b>Regular Orange Text</b> Command line output.
<b>■</b>	<b>Regular green text</b> Denotes the contents of a file or script.



## Part 16 - QmailAnalog

Qmailanalog performs some basic log analysis on those qmail log files and then outputs them to a desired location. In my case, I run qmailanalog every night and output the results to e-mail. Along with qmailanalog, we're going to install the "qlogtools" package. Qlogtools, as its name implies, provides an array of tools which can be used to analyze the qmail logs. We're going to use one of the qlogtool packages, tai64n2tai, to convert the timestamps on the log files from a machine readable format to a human readable format which will come to us when we get the finished report. After we've installed both Qmailstats and Qlogtools, we will create a script which you can run on a nightly basis to generate e-mail stats. The script will also incorporate qms-analog, which we installed with qmail-scanner previously. The qms-analog output will give add qmail-scanner stats to our nightly report.

First, let's install qmailanalog...

```
cd /downloads/qmailrocks/
```

```
tar zxvf qmailanalog-0.70.tar.gz
```

```
cd qmailanalog-0.70
```

```
make
```

```
make setup check
```

That's it. Qmailanalog is installed!

Now let's install qlogtools...

```
cd /downloads/qmailrocks/
```

```
tar zxvf qlogtools-3.1.tar.gz
```

```
cd qlogtools-3.1
```

```
mkdir /usr/local/man (if directory already exists, you're good to go)
```

```
make
```

```
./installer
```

OK. The qlogtools library of tools should now be installed.

Now we will implement a script to run Qmailanalog and then you can hook that script into the server's crontab to get stats generated every night.

The script below is a solid script that sends an email to the server administrator with both the qmailanalog output as well as qms-analog's readout of qmail-scanner's activities. Pretty sweet, huh?

```
cp /downloads/qmailrocks/qms-analog-0.4.0/qmailstats /var/qmail/bin
```

```
vi /var/qmail/bin/qmailstats
```

```

#!/bin/sh
## qms-analog and qmailanalog invocation script
##
## Note: For better readability of the nightly stats email, set your email
## client font to a fixed width font - then all the columns line up

```

```
## very nicely.
```

```
##
```

```
PATH=/usr/local/qmailanalog/bin:/var/qmail/bin:/bin:/usr/bin:/usr/local/bin
```

```
QMAILSTATS="/tmp/q.$$"
```

```
EMAILMSG="/tmp/qms.$$"
```

```
umask 077
```

```
DATE=`date +%D`
```

```
## prepare qmail log entries for qmailanalog routines
```

```
cat /var/log/qmail/qmail-send/* /var/log/qmail/qmail-pop3d/* /var/log/qmail/qmail-smtpd/* | tai64n2tai | awk '{ $1=substr($1,1,index($1,"")+6);print}' | matchup > $QMAILSTATS 5>/dev/null
```

```
## build the email message header
```

```
echo "To: your_postmaster@yourdomain.com" > $EMAILMSG
```

```
echo "From: your_postmaster@yourdomain.com" >> $EMAILMSG
```

```
echo "Subject: Nightly Qmail Stats Report for $DATE" >> $EMAILMSG
```

```
echo "" >> $EMAILMSG
```

```
echo "" >> $EMAILMSG
```

```
## qms-analog invocation
```

```
#
```

```
# USAGE: qms-analog hours-of-history <sort-key>
```

```
#
```

```
# hours-of-history (0 - n) hours of history to collect
```

```
# 0 => all records
```

```
# sort-key (optional) sort key for account statistics
```

```
# msgbw (default) msg bandwidth - successful msgs
```

```
# alpha alphanumeric by account name
```

```
# virus number of viruses received
```

```
# saavg Spamassassin avg score
```

```
# sadet Spamassassin msgs detected
```

```
#
```

```
# Examples:
```

```
# "qms-analog 24" - use only records within the last 24 hours,
```

```
# sort by msg bandwidth
```

```
# "qms-analog 168" - use only records within the last 7 days,
```

```
# sort by msg bandwidth
```

```
# "qms-analog 0" - use all records, sort by msg bandwidth
```

```
# "qms-analog 0 alpha" - use all records, sort alphabetically
```

```
# "qms-analog 0 saavg" - use all records, sort by Spam average score
```

```
#
```

```
# Note: Add or remove statistical time frames to suit your preference -
```

```
# "last 24 hours" and "all records" are uncommented below by default.
```

```
##
```

```
#### Last 24 hours
```

```
echo "~~~~~" >> $EMAILMSG
```

```
echo "~~~~~ L a s t 2 4 H o u r s ~~~~~" >> $EMAILMSG
```

```
cat /var/spool/qmailscan/qms-events.log | qms-analog 24 >> $EMAILMSG
```

```
####
```

```
#### Last 7 days
```

```
#echo "" >> $EMAILMSG
```

```
#echo "~~~~~" >> $EMAILMSG
```

```
#echo "~~~~~ L a s t 7 D a y s ~~~~~" >> $EMAILMSG
```

```
#cat /var/spool/qmailscan/qms-events.log | qms-analog 168 >> $EMAILMSG
```

```
####
```

```
#### Last 30 days
```

```
#echo "" >> $EMAILMSG
```

```
#echo "~~~~~" >> $EMAILMSG
```

```
#echo "~~~~~ L a s t 3 0 D a y s ~~~~~" >> $EMAILMSG
```

```
#cat /var/spool/qmailscan/qms-events.log | qms-analog 5040 >> $EMAILMSG
```

```
####
```

```

#### All records in qms-events.log
echo "" >> $EMAILMSG
echo "-----" >> $EMAILMSG
echo "----- A I I R e c o r d s -----" >> $EMAILMSG
cat /var/spool/qmailscan/qms-events.log | qms-analog 0 >> $EMAILMSG
####

## qmailanalog invocation
echo "" >> $EMAILMSG
echo "" >> $EMAILMSG
echo "-----" >> $EMAILMSG
zoverall < $QMAILSTATS >> $EMAILMSG
echo "" >> $EMAILMSG
echo "" >> $EMAILMSG
echo "-----" >> $EMAILMSG
zfailures < $QMAILSTATS >> $EMAILMSG
echo "" >> $EMAILMSG
echo "" >> $EMAILMSG
echo "-----" >> $EMAILMSG
zdeferrals < $QMAILSTATS >> $EMAILMSG
echo "" >> $EMAILMSG

## pipe the message into qmail-inject
cat $EMAILMSG | qmail-inject

## delete temp files
rm -f $QMAILSTATS
rm -f $EMAILMSG

```

Now set the script executable...

```
chmod 750 /var/qmail/bin/qmailstats
```

Now run the script...

```
/var/qmail/bin/qmailstats
```

Check your email and you should get a report with some pretty cool stuff in it! Your report should look [something like this](#).

OK, if the qmailstats script is working, you will now want to create a crontab entry to run this script every night.

So, as the "root" user let's set up a cron entry...

```
crontab -e
```

```
0 3 * * * /var/qmail/bin/qmailstats 1>/dev/null 2>/dev/null
```

Save and exit from the crontab editor and you should be all set. The above entry will run the qmailstats script every night at 3:00AM.

[Proceed to Part 17](#)

qmailrocks@earthlink.net

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Black Text</b>	Commands to be run by you, the installer.
■	<b>Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 17 - Installing Qtrap

Our final ingredient in this installation is going to be a domain level word filter, which I've named "Qtrap". This script is applied on a per domain basis and serves as a "bad word" scanner to catch any spam that Spamassassin may have missed. This filter serves as the last defense against SPAM before it arrived in your inbox. I like this filter because it helps to get rid of any SPAM that happens to make it by Spamassassin. Without any protection at all, my mailbox gets a shit ton of SPAM every day. Within the first 3 months I enacted the Qtrap filter, Qtrap logged over 9,000 deleted SPAM messages, none of which were legitimate e-mails. My keyboard's delete key was very appreciated the extra rest.

Any emails that are scanned and contain a banned word will be automatically deleted and logged by the qtrap script. A whitelist feature now exists so that individual addresses or domains can be exempt from the qtrap scan.

So let's install it...

```
cd /usr/home/vpopmail
```

```
mkdir -p qtrap/logs
```

```
cd qtrap
```

```
cp /downloads/qmailrocks/scripts/qtrap/qtrap-freebsd-2.0.0 ./qtrap.sh
```

Defining your whitelist:

```
vi qtrap.sh
```

You will see a block of code for the whitelist that looks like this:

```
whitelist_check () {
case $WHITELIST in
address@somewhere.com|address@somewhereelse.com|*entiredomain.com)
echo $SENDER found in whitelist on `date "+%D %H:%M:%S" >> /usr/home/vpopmail/qtrap/logs/qtrap.log
exit 0;;
*)
;;
esac
}
```

The email addresses in the **bold red text** above should be substituted with any email addresses that you wish to whitelist against the qtrap filter process. Whitelisted addresses will be allowed to send you mail that contains "banned" words. Un-whitelisted address will be scanned and their message deleted if it contains a banned word. As you can see above, you can specify an individual address (address@somewhere.com) or you can simply whitelist an entire domain (\*entiredomain.com).

Defining your "banned word" list:

within the qtrap.sh script you should see another section, below the whitelist section of code, that looks like this:

```
checkall () {
case $BANNED_WORDS in
porn|PORN|Sex|SEX)
printout $BANNED_WORDS
echo MESSAGE DROPPED from $SENDER because of $BANNED_WORDS on `date "+%D %H:%M:%S" >>
/usr/home/vpopmail/qtrap/logs/qtrap.log
exit 99;;
*)
;;
esac
```

}

The portion of the above section that I've highlighted in **RED** is the array of "banned" words. Edit this array to your satisfaction. Make sure that each word is separated by a pipe "|" and keep in mind that the array is case sensitive. So the words "SEX" and "Sex" are 2 different words. Also, exercise caution here. You don't want to ban words that are used in everyday e-mails. For example, you wouldn't want to ban the word "hello" or something like that. You should only ban words that you are 100% sure you would never see in a legitimate e-mail.

Now let's set up the logging directory...

```
touch /usr/home/vpopmail/qtrap/logs/qtrap.log
```

```
chown -R vpopmail:vchkw /usr/home/vpopmail/qtrap
```

```
chmod -R 755 /usr/home/vpopmail/qtrap
```

Now we will add this script into the mail path for a domain on our server.

```
cd /usr/home/vpopmail/domains/yourdomain.com
```

```
vi .qmail-default
```

add the following line above the line that is already there

```
| /usr/home/vpopmail/qtrap/qtrap.sh
```

Here's an example:

.qmail-default before:

```
| /usr/home/vpopmail/bin/vdelivermail " delete
```

.qmail-default after:

```
| /usr/home/vpopmail/qtrap/qtrap.sh
```

```
| /usr/home/vpopmail/bin/vdelivermail " delete
```

Save these changes and that should be it. You don't have to restart anything. To test this last rule, try sending an e-mail to your mailbox and make sure that the test e-mail contains one of the words that you entered into the "bad word" list in the Qtrap script. If the filter is working right, the message should NOT arrive in your inbox. You should then be able to view the log file at /usr/home/vpopmail/qtrap/logs/qtrap.log and see a log of the dropped message corresponding to the time at which you sent the test message. The drop log should look something like this:

```
MESSAGE DROPPED from someone@somewhere.com because of some_banned_word on 06/13/03 02:37:51
```

If the test was successful, then that's it! Congratulations, you've completed the Qmailrocks.org Qmail installation. Have fun. The next couple steps discuss cleanup as well as some closing notes and suggestions.

[Proceed to Part 18](#)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
■	<b>Black Text</b>	Commands to be run by you, the installer.
■	<b>Maroon Text</b>	Special notes for Redhat 9 users.
■	<b>Red Text</b>	Vital and/or critical information.
■	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text	Command line output.
■	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 18 - Maintaining your qmail server

Once you've got your qmail server up and running, how do you take care of it? This page will cover the many answers to that question. So here goes.

Table of contents:

[Making sure that all services start on boot](#)  
[Maintaining the qmail queue](#)  
[Maintaining qmail-scanner](#)  
[Maintaining SpamAssassin](#)  
[Maintaining Clam Antivirus](#)  
[Maintaining current software versions](#)  
[Maintaining the qmail logs](#)  
[Maintaining administrative mailboxes](#)  
[Maintaining other mailboxes](#)  
[Backing up your qmail server](#)

### Making sure that all services start on boot

If you've installed qmail correctly, it should already automatically start when you boot your server. However, you will want to make sure that all of the other needed services start as well.

#### --For Redhat users--

Starting Courier-imap on boot - make sure the following 2 lines exist in your server's /etc/rc.local file:

```
/etc/rc.d/init.d/imap start  
/etc/rc.d/init.d/imaps start
```

Starting SpamAssassin on boot

Try running the "setup" command and check the system services area to see if SpamAssassin is listed and marked with a "\*" to start on boot. If SpamAssassin is not present in the system's "setup" tool, you can start it on boot by added the following line to the server's /etc/rc.local file

```
/etc/rc.d/init.d/spamd start
```

Starting Apache on boot

Try running the "setup" command and check the system services area to see if the "httpd" service is listed and marked with a "\*" to start on boot. If it is not present in the system's "setup" tool, you can start it on boot by added the following line to the server's /etc/rc.local file

```
/etc/rc.d/init.d/httpd start
```

#### --For Slackware users--

Make sure the following startup commands exist in your system's /etc/rc.local file:

Starting Courier-imap

```
/etc/rc.d/rc.imap start  
/etc/rc.d/rc.imaps start
```

Starting SpamAssassin

```
/etc/rc.d/rc.spamd start
```

Starting Apache

```
/etc/rc.d/rc.httpd start
```

 (If you already have Apache configured to start on boot, you can omit this line)

## Maintaining the qmail queue

It's usually not a bad idea to keep your qmail server's queue in check. Your qmail server's queue is located at `/var/qmail/queue`. However, it's just about impossible to look directly at the queue folders and be able to tell what's going on.

**qmailctl stat** - This, of course, shows you your qmail server's current status. Included in the stats is the current condition of the queue. You can use this to see how many messages are sitting in the queue.

**qmHandle** - This add-on tool allows more in depth viewing of the queue and it also allows you to perform administrative functions on the queue. You can find instructions about installing qmHandle [here](#). Once qmHandle is properly installed, running the command "qmHandle" will provide a listing of all the possible commands possible. I'll go over a couple of them right now. The "qmHandle -l" command will give you a complete listing all every message in the queue and a summary containing the date the message was sent, the sender and the intended recipient. The output for a single message might look like this:

```
6406395 (195, R)
Return-path: bob@somewhere.com[]
From: Bob Smith <bob@somewhere.com>
To: Frank Smith <frank@wherever.com>
Subject: Re: This weekend
Date: Mon, 16 Feb 2004 12:14:31 -0700
Size: 1482 bytes
```

The message number, 6406395, can be seen at the top of the message. If we wanted to delete this single message from the queue, we could do so with a command of "qmHandle -d6406395". The entire queue can be cleared out with the command "qmHandle -D".

**queuelifetime** - The "queuelifetime" setting for qmail determines how long messages can stay in the queue. By default, your qmail server will keep messages in the queue for 604,800 seconds, or 7 days. However, you can set a custom queuelifetime by creating a file called `/var/qmail/control/queuelifetime`. The content of that file is a single line containing a number which represents the number of seconds the queue will hold any given message. If you want to set a custom queuelifetime, you might want to whip out the old calculator.

## A little advice on handling your queue

You need to keep in mind that a queued message is NOT necessarily a BAD thing. The only time a message is queued for any length of time is when it is undeliverable at the time it is originally sent. A message is usually undeliverable for 1 of 2 reasons:

**1)** The receiving mail server is offline. If this is the case, when that mail server comes back online at a later time your qmail server will then be able to deliver that message. A perfect example of this is the Qmailrocks.org mailing list. At any given time there lots of messages in the mailing list server's queue. However, these messages usually get delivered eventually. As you can imagine, some people on the mailing list are probably using a new qmail server to house the mailbox with which they have signed up on the list with. Well, of course, since that person is new the qmail there is a chance that their server may go down for periods of time while they are working on it and perfecting their qmail skills. It is at these times when their server is unreachable and when my list server then queues the message for later delivery. No big deal.

**2)** The recipient address is bogus or incorrect. If you get hit with a lot of spam on your server, or if you have a spammer on your server, this will probably happen to you at some point. Your queue gets filled with message bound for bogus addresses OR it gets filled with bounce messages that your server is trying to deliver back to spammers who sent spam to bogus addresses on your server. A good way to cut down on this is to set the domains on your server to "delete" catchall mail instead of bouncing it. This can be done from within the Qmailadmin interface. Setting a domain's catchall setting to "bounce" is a bad idea in my experience as it only results in a queue full of bounced bounce messages. If you don't need a catchall for your domain, do yourself a favor and set it to "delete". This is also the case if you are running Qmail-scanner. Qmail-scanner has an option to notify the sender when a virus is found in an e-mail. Bad idea. Most of the time, the address from which that virus laden e-mail came is bogus. So trying to "notify" the sender usually results in nothing but a bunch of double bounces flying all over your server.

Fortunately, since your qmail server has a built in queue lifetime, messages will eventually drop out of the queue if they are undeliverable. The last thing I want to mention is that it is a common misconception that if your queue is full, mail being sent presently will not be delivered quickly. This is a misconception. As I said, the queue is a repository for messages that are not immediately deliverable. If a message is being sent to a valid address it will get sent immediately, regardless of the size of the queue. So, in summary, you don't need to panic every time you have messages in your queue.

Most of the time, the best thing to do is to just let your queue take care of itself. However, there are time when you may deem it necessary to clean out the queue or take other administrative action, and that's what the "qmHandle" tool above is good for.

## Maintaining qmail-scanner

Qmail-scanner is fairly easy to maintain. Once you get it configured how you want it to be, the main items your going to want to worry about are 1) the log files and 2) the virus quarantine area. Here's some info on both.

**1) qmail-scanner logs** - Qmail-scanner, when Clam Anti Virus and SpamAssassin are hooked into it, logs the virus scanning activities to `/var/spool/qmailscan/qmail-queue.log`. This log file can get REALLY big, so you will want to keep it in check. You may want to set up a rotation schedule for this log file or some other sort of scheduled task that deals with this log file on a routine basis. If this log file exceeds the linux file size limit of 2GB, your mail server will start freaking out and all hell will break lose. So do yourself a favor and keep an eye on this log file.

**2) The virus quarantine area** - When qmail-scanner pipes the mail out to Clam Anti Virus and virus is found, the virus laden message is quarantined at `/var/spool/qmailscan/quarantine/new`. Those e-mails will usually not pose any threat to your server, since there are very few Linux/Unix viruses and since they are not being executed. However, on a busy mail server, that folder can get filled up pretty quickly, so you may want to keep an eye on it and have it emptied on a routine basis. I empty mine out with a crontab that runs once a week.

Anytime you upgrade qmail-scanner, it's a good idea to refresh the qmail-scanner perlscanner database:

Redhat/Fedora/RHEL:

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -g or /var/qmail/bin/qmail-scanner-queue -g (for non setuid setups)
```

Debian:

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -g or /var/qmail/bin/qmail-scanner-queue -g (for non setuid setups)
```

Slackware:

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -g
```

## Maintaining SpamAssassin

SpamAssassin is relatively maintenance free one you get it up and running. It logs it's activities to `/var/log/maillog`, so you can always reference the logs for any investigations. New versions of SpamAssassin are released fairly often, so you may want to occasionally check <http://www.spamassassin.org> to see if there have been any new releases. In my experience, I've always been able to install the new version over the older version with no problems. If you upgrade, just make sure the `/etc/mail/spamassassin/local.cf` file still contains the setting you want and you should be in good shape.

Anytime you upgrade SpamAssassin, it's a good idea to refresh the qmail-scanner perlscanner database:

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -g or /var/qmail/bin/qmail-scanner-queue -g (for non setuid setups)
```

## Maintaining Clam Antivirus

The only things you really need to do to maintain Clam AV are:

**1)** Make sure your server is running the automatic virus definition updates on a regular basis. I run the following command out of crontab on a routine basis:

```
/usr/bin/freshclam --quiet -l /var/log/clamav/clam-update.log
```

**2)** Keep your version relatively current. New releases are put out fairly often, so keep an eye on it. Again, in my experience I've been able to install the newer version over the older one with no problems.

**3)** Anytime you do decide to upgrade Clam Antivirus, you will need to update qmail-scanner's version file. This is easily done by running the following command:

```
setuidgid qscand /var/qmail/bin/qmail-scanner-queue.pl -z or /var/qmail/bin/qmail-scanner-queue -z (for non setuid setups)
```

## Maintaining current software versions in general

Naturally, as time passes new and improved versions of software will be released from their respective venders. It's always nice to have to latest

versions of everything, but unless the newer version fixes a security hole or a major bug you don't need to lose sleep over it. You can check the vendors' websites every now and then to see if a new version is out. I try to keep the QMR software package as current as possible and I'm pretty good at it. Most of the time, the qmailrocks.tar.gz package will contain the latest versions of everything.

I've never had any big problems with upgrading any of the software. For source packages, you can usually simply compile and install the latest version right over the older version with no problems. I mention this in the above paragraphs on SpamAssassin and Clam AV, but it generally applies to all of the software found on this site. The same goes for RPMs. You can usually just install the new RPM right over the older one using the command syntax "rpm -Uvh whatever.rpm".

Qmail itself has not had a new version release in a very long time, but you can bet if there is a new release I will have it here. As you probably know by this point, the current qmail version is version 1.03. Many people have noticed that qmail.org offers a version of qmail called "netqmail-1.05". At first glance this may appear to be a newer version of qmail. IT IS NOT. Netqmail is simply qmail-1.03 with some of the patches (which I use on this site anyway) already built in. If you follow the QMR installation guide using 1.03, the resulting installation of qmail is not different that if you used netqmail. Understand? If you don't trust me, check out the full description of netqmail [here](#).

## Maintaining the qmail logs

Fortunately, qmail's logs take care of themselves. They automatically rotate all on their own, so you never have to worry about them. The only thing worth noting is that you CAN customize the rotation schedule for the logs. This is done the "logs" supervise script for each supervised function. Confused? Ok, I'll explain.

In the QMR qmail installation, there are 3 supervise scripts and, subsequently, 3 logs for those scripts.

`/var/qmail/supervise/qmail-pop3d/run` operated the pop server and is logged via `/var/qmail/supervise/qmail-pop3d/log/run`

`/var/qmail/supervise/qmail-smtpd/run` operates the smtp server and is logged via `/var/qmail/supervise/qmail-smtpd/log/run`

`/var/qmail/supervise/qmail-send/run` operates the mail processor and is logged via `/var/qmail/supervise/qmail-send/log/run`

Each of the above "log/run" scripts tells the server how it wants those activities logged. Let's take the `/var/qmail/supervise/qmail-pop3d/log/run` logging script as an example:

```
#!/bin/sh
PATH=/var/qmail/bin:/usr/local/bin:/usr/bin:/bin
export PATH
exec setuidgid qmail multilog t s100000 n20 /var/log/qmail/qmail-pop3d 2>&1
```

The last line of this script sets some of the logging options. We can break that last line into several parts:

```
exec setuidgid qmail multilog t s100000 n20 /var/log/qmail/qmail-pop3d 2>&1
```

So let's break it down:

`exec setuidgid qmail multilog` - run the multilog program as the "qmail" user.

`t` - inserts an @, a precise timestamp, and a space in front of each line.

The above "t" is why the rotated logs take on names like this: `@40000000402d1c562cbf3534.s`

`s100000` - logs will rotate when they reach 100000 bytes.

`n20` - number of rotations to keep on hand before they fall off.

`/var/log/qmail/qmail-pop3d 2>&1` - the directory to where the logs will be written, silently.

## Maintaining administrative mailboxes

This may seem a bit silly, but you'd be surprised how many people neglect the administrative e-mail account on their server. But what is the administrative e-mail account? Well, that depends on you. Administrative mail would be mail destined for the server's root user as well as any notification settings you may have. The destination for most the server's administrative addresses is usually determined by the aliases you have created at `/var/qmail/aliases`. If you've followed the QMR install guide, you should have a "postmaster" alias as well as "root", "mailer-daemon" and

"anonymous" aliases. If you're like me, you direct all of these aliases to a single mailbox on your qmail server. For example, on the qmailrocks.org server, I direct all of these aliases to the qmailrocks.org postmaster account. This means that my postmaster account gets all of the administrative mail on the server. I get bounce failure messages, crontab reports, log watch reports and many other admin type emails. In addition, since I also have qmail-scanner set up to send virus reports to this same address. So, as you can see, my postmaster account is the central locus for all the server's administrative mail. You probably want to do the same with your server, as these administrative e-mails can often help you to find and correct problems that might otherwise go unnoticed. A mistake that people make a lot is to have the administrative mail directed to some mail account that they never check. This inevitably leads to that person being surprised when they find out that they have a mailbox on their server that's a couple Gigs in size. These people will also be surprised when they find that that mailbox is full of error messages that have been coming in for months indicating that something on the server is misconfigured. They never checked the mailbox, so they never knew. Well, I guess ignorance is bliss. So my point of this whole paragraph is that you should keep your administrative mail configuration organized and well cared for. It will save you a lot of heartache down the road.

### Maintaining other mailboxes

Managing all other mailboxes on your server is made easy my simply setting quotas on all domains. It is inevitable that if you host mail on your server, there will be some idiot who either never checks his mailbox or decides that he has to store a lifetime worth of mail on the server. Setting quotas for your domains is a way of keeping these idiots in check and preventing you from having a disk space crisis because of these idiots. If you have a 120GB drive in your server and no quotas, there WILL be some fool who fills up all 120GB with his mail. So take my advise and set rigid quotas for your domains.

### Backing up your qmail server

Backing up a qmail server is relatively easy. While different people may give you slightly different recommendations, you can ensure a safe backup of your qmail server if you backup the following 2 directories on a routine basis.

**/usr/home/vpopmail** - backs up all your domain information, including mailboxes, passwords and the messages themselves.

**/var/qmail** - backs up all of your qmail settings. The **/var/qmail/control** directory is the most important directory in there to back up, but it won't hurt to just back up the whole damn qmail directory.

[Proceed to Part 19](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text   Qmail installation notes and summaries by the author. Me talking.
<b>Black</b>	<b>Bold Black Text</b>   Commands to be run by you, the installer.
<b>Maroon</b>	<b>Bold Maroon Text</b>   Special notes for Redhat 9 users.
<b>Red</b>	<b>Bold Red Text</b>   Vital and/or critical information.
<b>Purple</b>	<b>Regular/Bold Purple text</b>   Denotes helpful tips and hints or hyperlinks.
<b>Orange</b>	<b>Regular Orange Text</b>   Command line output.
<b>Green</b>	<b>Regular green text</b>   Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 19 - Mail Client Configuration

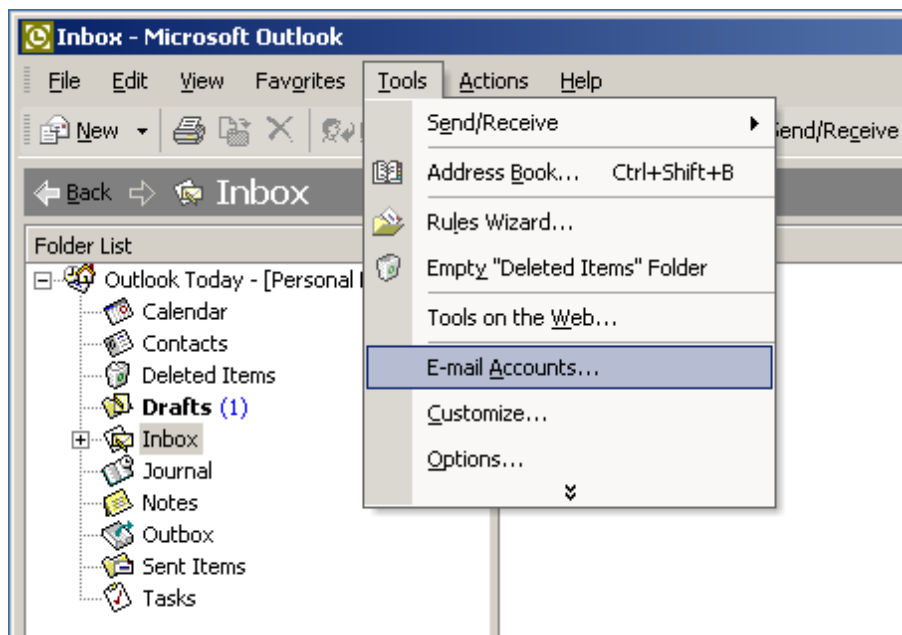
So now you've got a fully functioning mail server. Well, now it's time to make use of that mail server and configure a remote mail client to retrieve mail from the server and send mail through the server. For my example, I've chosen Microsoft's Outlook XP. Outlook Express varies slightly, but you should be able to get the general idea. Likewise, if you using some other mail client you should be able to get the idea. Please do not e-mail me asking for instructions on another mail client. This is the only one I'm going to provide for now.

Anyway, let's configure Outlook...

For this example, let's set the following pre-conditions:

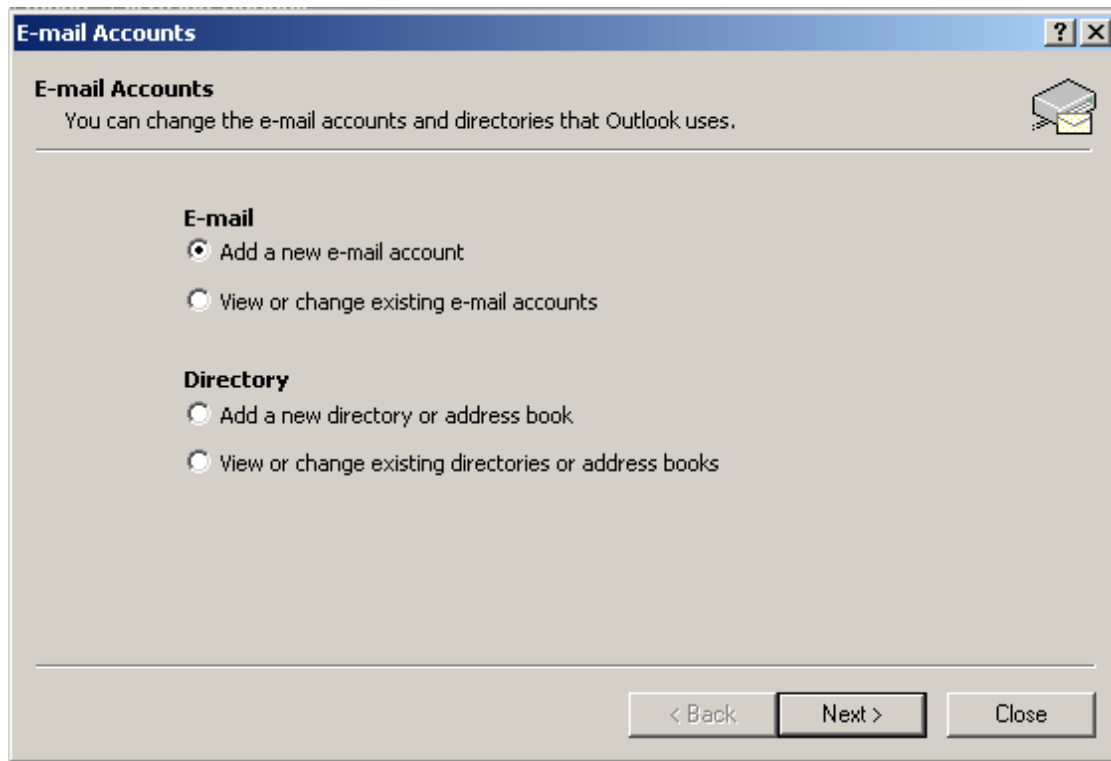
1. I have used Vqadmin to set up a new domain on my server called "domain.com"
2. Since I have set the domain up using Vqadmin, there should already be a main user for the domain called "postmaster@domain.com".
3. I'm going to set up Outlook to check mail for and "postmaster@domain.com" and will also use that user for SMTP authentication, so that I can send mail.

### Step 1: Go to E-mail account configuration



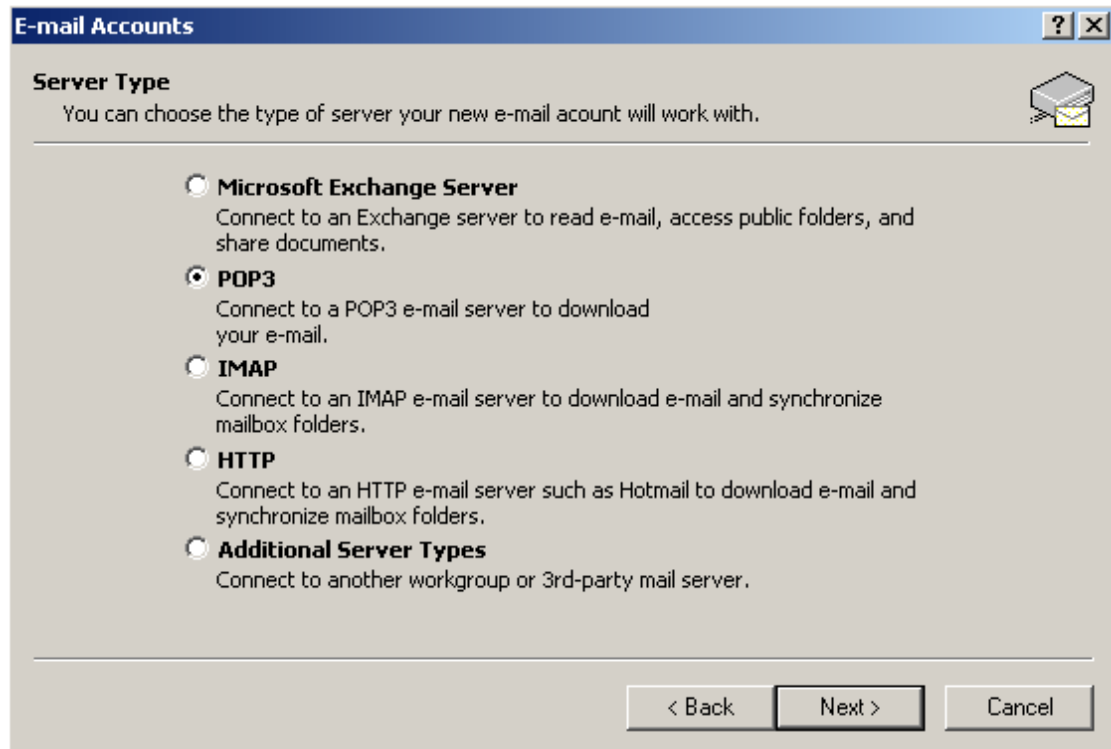
Select "E-mail Accounts" under Tools.

### Step 2: Add a new E-mail account



When finished, click "Next".

### Step 3: It's a POP3 type of account



When finished, click "Next".

### Step 4: Configure the account

**Click on "More Settings" once you've filled out the above information.**

Step 5: Enable SMTP Authentication (under the "more settings" area):

**Save all changes and give it a go! You should be all set. Now that wasn't so hard, was it?**

**SMTP Settings Troubleshooting Tip**

You may find that when you configure Outlook to use your Qmail server as the outgoing SMTP server, those connection attempts get blocked. You may get an error message like "Could not connect to server". Often, this is due to your ISP blocking foreign connections to port 25 (smtp). Most large ISP's these days will block attempts to connect to a foreign SMTP server. This is an anti-spam measure. The solution is to use your ISP's SMTP server instead of your own.

Let's take a real-life example:

Earthlink, for example, blocks connection attempts to foreign SMTP servers. I use Earthlink DSL, therefore I have to configure Outlook to use Earthlink's SMTP servers (mail.earthlink.net) instead of my Qmail server's. So, in the SMTP server setting in Outlook, I stick in "mail.earthlink.net" instead of my server's address. Of course, whether or not you configure Outlook to use SMTP authentication at this point is strictly determined by what your ISP's SMTP connection rules are. If you need help, contact your ISP.

[Proceed to Closing Notes](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text Qmail installation notes and summaries by the author. Me talking.
<b>Black</b>	<b>Bold Black Text</b> Commands to be run by you, the installer.
<b>Maroon</b>	<b>Bold Maroon Text</b> Special notes for Redhat 9 users.
<b>Red</b>	<b>Bold Red Text</b> Vital and/or critical information.
<b>Purple</b>	<b>Regular/Bold Purple text</b> Denotes helpful tips and hints or hyperlinks.
<b>Orange</b>	<b>Regular Orange Text</b> Command line output.
<b>Green</b>	<b>Regular green text</b> Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**



## Part 20 - Feedback

Please take a minute to provide me with some feedback...

How helpful did you find this site?

This site stinks!    Not very good    It was ok    Very helpful    Excellent

Your Name:

(if you want me to reply)

Your E-mail:

(if you want me to reply)

Any comments/suggestions?

May we post your comments on this website?    Yes    No  
 (your name will not be posted)

Would you like to join the [mailing list](#)?    Yes    No

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key	
	Regular Black Text    Qmail installation notes and summaries by the author. Me talking.
■	<b>Bold Black Text</b> Commands to be run by you, the installer.
■	<b>Bold Maroon Text</b> Special notes for Redhat 9 users.
■	<b>Bold Red Text</b> Vital and/or critical information.
■	Regular/Bold Purple text    Denotes helpful tips and hints or hyperlinks.
■	Regular Orange Text    Command line output.
■	Regular green text    Denotes the contents of a file or script.



## Part 5 - Vpopmail w/mysql

Vpopmail is one of the major components of this installation. Vpopmail allows us to do virtual domain mail hosting. It's got a lot of built in tools and features that make it a dream to work with. Even if you don't want to host mail for multiple domains, I would still recommend installing Vpopmail. It just makes the whole mail game easier. Plus, my installation centers around it, so if you don't install it you're going to have a headache.

This page details the configuration and installation of vpopmail with MySQL functionality. When used with mysql, all domain information (domain name, email users, passwords etc.) are all stored in a handy mysql database. The actual mail messages are NOT stored in the database and are still kept in the same areas as a standard vpopmail build. I would not recommend integrating mysql into vpopmail if you're only going to be hosting a few domains on your email server, but if you plan on hosting great numbers of domains then the mysql component is a practical solution. It also makes it easy to port your mail server to another machine if you ever have to upgrade or recover from a crash.

**A note for NEWBIES:** If you are a newbie to qmail and especially to mysql, I would strongly recommend that you NOT mess with this section and that you simply install a [standard build](#) of vpopmail. I'm not going to hold your hand on how to install mysql, how to configure mysql and how to administer a mysql server. You should only proceed with this section if you are very comfortable with mysql. I will ignore any and all newbie related questions to this section. In general, on a linux build, you will want to have both mysql-server and mysql-devel installed (along with any dependencies), but that's all I'm gonna say on the matter.

So let's install it...

If you recall, we already created a "vpopmail" system user and a "vchkpw" system group in step 2 when all the needed users and groups were created. So, now we will set up the vpopmail mysql connection config file under the vpopmail user's home directory.

```
mkdir ~vpopmail/etc
```

```
chown vpopmail:vchkpw ~vpopmail/etc
```

Obviously, you will want to substitute in your own vpopmail username and password below. You can make the username and password anything you wish.

```
echo "localhost|0|vpopmailuser|password|vpopmail" > ~vpopmail/etc/vpopmail.mysql
```

Now we set the proper ownership/permissions on the new mysql connection config file...

```
chown vpopmail:vchkpw ~vpopmail/etc/vpopmail.mysql
```

```
chmod 640 ~vpopmail/etc/vpopmail.mysql
```

The next step is to log into your mysql server as the "root" mysql user and create the vpopmail database as well as a vpopmail user with proper rights on that new database.

```
mysql -u root -p
```

Enter your mysql server's root password when prompted.

And now we create the new vpopmail database...

```
CREATE DATABASE vpopmail;
```

And we create a vpopmail user that will have access to that database.

**The username and password that you create here MUST match the information that you entered in the config file above.**

```
GRANT select,insert,update,delete,create,drop ON vpopmail.* TO vpopmailuser@localhost IDENTIFIED BY 'password';
```

And now let's quit...

**quit**

And let's test the new user we just created...

**mysql -u vpopmailuser -p**

Enter the vpopmail user's password when prompted. If you get in, you're golden. If you are denied access, you screwed up. Go back and repeat the above steps.

Ok, so that does it for the mysql setup portion. Now let's configure vpopmail to actually use the mysql functionality and then install it.

**cd /downloads/qmailrocks**

**tar zxvf vpopmail-5.4.7.tar.gz**

**cd vpopmail-5.4.7**

**./configure --enable-logging=p --enable-auth-module=mysql --disable-passwd --enable-clear-passwd --disable-many-domains --enable-auth-logging --enable-mysql-logging --enable-valias --disable-mysql-limits**

**make && make install-strip**

If you don't get any errors, then Vpopmail is good to go! Further ahead in the installation, we'll create domains under vpopmail. At that time, if you wish, you can log back into mysql and confirm that the mysql entries have been created for the domain(s) you add.

**Wait! I'm getting an error! What do I do?**

If you are using GCC version 3.x, you may get a compilation error similar to this:

**In file included from vconvert.c:35:  
 vmysql.h:53:22: missing terminating " character  
 vmysql.h:60:35: missing terminating " character  
 make[2]: \*\*\* [vconvert.o] Error 1**

If you get this error, you will need to apply a patch to Vpopmail:







**patch < /downloads/qmailrocks/patches/vmysql.patch**

Once you've applied the patch, try running "make" and "make install" again and you should be ok.

So now let's go on to part 6...

[Proceed to Part 6](#)

[qmailrocks@earthlink.net](mailto:qmailrocks@earthlink.net)

Color Coded Qmail Installation Key		
	Regular Black Text	Qmail installation notes and summaries by the author. Me talking.
	<b>Bold Black Text</b>	Commands to be run by you, the installer.
	<b>Bold Maroon Text</b>	Special notes for Redhat 9 users.
	<b>Bold Red Text</b>	Vital and/or critical information.
	Regular/Bold Purple text	Denotes helpful tips and hints or hyperlinks.
	Regular Orange Text	Command line output.
	Regular green text	Denotes the contents of a file or script.

[home](#) | [about](#) | [the installation](#) | [scripts & utilities](#) | [faq](#) | [contact](#) | [links](#) | [donations](#) | [journal](#) | [mailing list](#) | [list archive](#) | [qmr forum](#)

Site last modified: **December 05, 2004 23:29:41 EST**

