

# How to use Spamassassin together with Qmail

Georg Lehner

2 de febrero de 2003

Although (by now) I don't use Spamassassin myself, I have it done once and it worked for me. People have asked me about it, so I put the pieces here together. As with most of the pages on this site you should be able to download it in PDF format.

## 1. Software

Qmail is a very secure and fast and complete and... mailserver software. The home page, maintained by it's author Dan Bernstein is terse (but very good), if you look for abundancy look at [www.qmail.org](http://www.qmail.org). A very popular description for setting up Qmail is Live with Qmail. My favorite installation method however is under Debian GNU/Linux with Gerrit Papes unofficial binary packages.

Spamassassin is a filter for Internet Mail Messages, which does Header analysis and real language matching, to get hints if a given message has to be considered Spam. Each filter rule is weighted with a score, if a certain score is reached – the threshold – the mail is considered Spam. Messages that *are* spam can get through spamassassin untouched, and messages that are *not* spam can be tagged as such. The continuous refinement of the filter rules, and some accompanying methods try to improve the detection rate. I suppose, spamassassin is one of the very good spam filters.

ifspamh is a /bin/sh script written by James Grinter to simplify mail filtering with Spamassassin.

## 2. Considerations

I don't like spamfilters:

- The rules have to be adapted constantly as new forms and content variations of spam are developed. Filters always lag behind.
- Each mail on the system (or of the user who runs it) has to be filtered by a quit heavy filter. I run small and “slow” servers and have typical filter times of 3 to 25 seconds. Once my server got a lot of messages at once and got busy for half a day...
- It is not the way how to fight the evil. Look at the Im2000 Project or at my small contributions to it, to find out what’s completely wrong with Internet Mail, and how some people are trying to fix it.
- I always forget, that Spamassassin has a two times a double ess in it’s name, so my scripts always fail.

The best and most compact introduction to Spam reduction I found on the net is Chris Hardie’s page on the subject.

When you plan to use Spamassassin you have to decide if you:

- install it as a single user for individual mail filtering
- install it as a system wide Spam filter.
- only tag the messages, but let them all through.
- throw away messages which are considered to be spam by Spamassassin.

Chris’s page can help you decide what fit’s you best.

Furthermore, spamassassin can be installed as a daemon, than you use a small client program to send messages through the daemon and get the filtered message back. This helps with response time per message, as spamassassin has not be loaded from disk into memory each time a message gets filtered but only once at startup. You might however consider the increased static load a burden on small systems, a waste on fast systems with low mail volume (typical Workstation), and it can be considered as an increased security risk.

### **3. Filtering during mail delivery**

In the first place, I only describe message tagging. If you want to block spam hits you’ll look at the spamassassin documentation which tells you the right commandline options.

Filtering mail with spamassassin per user is accomplished by putting the line:

```
| spamassassin -P |maildir ./Maildir/
```

into `.qmail` in your home directory. You need `maildir`, which is part of the `safecat` package. The mail is piped into `Spamassassin`, `-P` tells it to write the filtered and eventually tagged message to `stdout`, rather than to deliver it to the mail spool. `maildir` puts it where it would have gone anyway with a decent `Qmail` installation, into the default `maildir` directory.

Now as to system wide delivery, you can change the default local delivery method. If you use the “Live with `Qmail`” setup, or modify Gerrit Pape’s installation to do this, the you will find a file `/etc/qmail/control/defaultdelivery`. This file maybe contains the two lines:

```
| dot-forward .forward
./Maildir/
```

Now you’ll change it to:

```
| spamassassin -P | dot-forward .forward
| spamassassin -P | maildir ./Maildir/
```

Maybe you’ll leave “`spamassassin`” out in the first line, because you consider that if spam is forwarded to a local user account it is checked by the second rule later on and you spare the double invocation of `Spamassassin`, bad luck for them if it goes to a `unspamassassined` location.

This method has the feature, that a user can override spam checking by setting up a `.qmail` file. Also outgoing mails are not checked, only mails locally delivered.

## 4. Filtering during queue processing

If you want to check/tag *all* mail going through your server, you need to do it when mail is queued. The original `Spamassassin` instructions tell you to apply the `QMAILQUEUE` patch to your `Qmail` programm. This will cause that the program `qmail-queue` when invoked will look at the environment variable `QMAILQUEUE`, if it is set, the contents is taken as a program which will be executed as a filter *before* piping the corresponding message through `Qmail`s `qmail-queue`. So you have to manage, that this environment variable is set to “`spamassassin -P`” at the right moments.

Another way is to set up a small script which invokes the filter/spooling chain and put it into `qmail-queue`’s place. The script should look similar to this one:

```
#!/bin/sh
spamassassin -P | qmail-queue.orig
```

After stopping Qmail, you go to the `/var/qmail/bin` directory, and rename `qmail-queue` to `qmail-queue.orig`.

```
cd /var/qmail/bin; mv qmail-queue qmail-queue.orig
```

Then you copy the script into place, maybe like we did in this example:

```
cp spamassassin-script /var/qmail/bin/qmail-queue
```

The script will be run as the `qmaild` user, and we need to create a `.spamassassin` directory at this users home:

```
mkdir ~qmaild/.spamassassin
chown qmaild:nofiles ~qmaild/.spamassassin
chmod u=rwx,g=rx,o= ~qmaild/.spamassassin
```

`qmail-queue`, when invoked from `qmail-smtp`, is often restricted to a memory limit, for example here is Gerrit Pape's run script for `/service/qmail-smtpd`:

```
#!/bin/sh
QMAILDUID='id -u qmaild'
NOFILESGID='id -g qmaild'
MAXSMTPD='cat /var/qmail/control/concurrencyincoming'
exec softlimit -m 2000000 \
    tcpserver -v -R -l 0 -x /etc/qmail/tcp.smtp.cdb -c "$MAXSMTPD" \
        -u "$QMAILDUID" -g "$NOFILESGID" 0 smtp qmail-smtpd 2>&1
```

the line with `exec softlimit -m 2000000` tells `tcpserver`, to limit all subprocesses to a maximum memory consumption of two (2) million bytes. If this results in too little memory the `spamassassin` process may die. Yury Gusev is using (by now) `exec softlimit -m 66000000`. Please tell us your experiences. It should also be considered to run the `spamassassin` daemon, and only invoke `spamc`, the `spamassassin` client with this setup, instead of launching a whole `spamassassin` process for each mail.

After realizing the respective adjustments to your system you start Qmail again.

## 5. Where is the “maildir” program?

`maildir` is part of “safecat”, a program that stores data received on the `stdin` safely to the disk, using Dan Bersteins `Maildir` algorithm.

You can

```
apt-get install safecat
```

it in a standard Debian/GNU/Linux installation or get it from the authors web-page.

Recently it seems that Debian's package does not provide "maildir", which is a simple wrapper around safecat. You can use:

```
| spamassassin -P | safecat ./Maildir/tmp ./Maildir/new
```

or create "maildir" by yourself:

```
#!/bin/sh # WARNING: This file was auto-generated. Do not edit!  
# Copyright (c) 2000, Len Budney. See COPYING for details.  
exec \  
/usr/local/bin/safecat "$1"/tmp "$1"/new  
# Change this to your safecat's path!
```

remember to put it into a place, reachable by .qmail's path.

## 6. Spamassassin close up

When spamassassin thinks it found a spam-mail, it can act in two ways, only notify with some statuscode that "this mail could be spam", or (additionally) change the message, so that the user can see it. The second option is called message tagging.

The default procedure for tagging a supposed to be spam message is to change the subject line in a way that it tells you it is spam and adding some lines to the body of the message, where the proves for the claim are given. Attachments are mime-defanged, i.e. they mime-type is changed to a "harmless" type, so the mail reader program will not accidentally open an application with the contents of the probably virus loaden content.

All these behaviour can be changed in the configuration. Some people only want a header "X-Spam:" added, which get's invisible when reading the message, but can be used for putting the mail into a "Spam" folder. Others want to stop delivering to a local mailbox at the moment a suspected Spam is dedected – in this case no modification of the mail is done, but the output status of the filter command is taken as a conditional for mail delivering.

## 7. Credits

Go of course to the authors of all that prime-quality software.

My readers have helped me to extend this document. Special thanks go to Yura Gusev for trying out the qmail-queue substitute method, and troubleshoot it.