DeletingAllMailsMarkedSpam

Is it sensible to simply delete all mail that SpamAssassin marks as being spam?

In general, no. While SpamAssassin is very good at picking out a large proportion of spam, it's impossible for a computer to do this job perfectly. Legimate mail incorrectly marked as spam is known as FalsePositives or FPs (see AvoidingFpsForAdmins and AvoidingFpsForSenders). You should only delete mail if you (and your users/customers) would find it acceptable to lose mail that are FPs. A much better idea is to filter possible spam into a separate folder that can be checked less frequently than the normal mailbox.

It is possible to reject the mail at the smtp level, generating a delivery error, so the sender is notified that their message is rejected. You need to use a mail server that supports this, such as mimedefang+sendmail. Many more are listed in IntegratedInMta. Mimedefang also allows you to save the mail to a central archive that you can extract from if you get an FP. If you do reject mail at the 5xx delivery level you need to set your spam threshold higher than the default of 5.

But I really really want to do it anyway!

Don't say we didn't warn you 😉

SpamAssassin itself will not delete any emails. It's only a filter which reads email in, and passes that same email out, modified in some way. If you want to delete emails, or redirect emails, you need to do it in whatever program calls SpamAssassin.

The following procmail script will delete mail with a score of 15 or higher. By moving the # (comment) mark up one line, it will save all mail with a score of 15 or higher in a separate folder rather than deleting. The general implementation then, is that mail with a score of less than 5 goes into the inbox, a client side rule triggering on "X-Spam-Status: YES" will cause mail scoring between 5 and 15 to go into a Junk Mail folder where it can regularly be checked, and mail of 15 or higher will be stored on the server or dropped on the floor.

```
Uncomment the following 3 lines and use tail -f procmail.log to debug
#LOGFILE=$HOME/procmail.log
#VERBOSE=yes
#LOGABSTRACT=all

# Send all mail through SpamAssassin

:0fw: spamassassin.lock
* < 256000
| spamassassin


# Mail that is very likely spam (>15) can be dropped on the floor.
# Move the # up one line to save it on the server instead.
# Note that dropping mail on the floor is a *bad*
# idea unless you really, really believe no false positives will
# have a score greater than 15.

:0
* ^X-Spam-Level: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
/dev/null
#almost-certainly-spam
```

A more feature-filled version of this script than supports learning and forwarding is at ProcmailToForwardMail.

How can I configure procmail to bounce messages flagged as spam?

Don't do this. Most spam forge the From line. If you bounce to it, you'll likely just be contributing to the Joe Job of some innocent soul. If you want to bounce messages (so that senders know it didn't reach you), you need to do it at the MTA level as described above.

Qmail

If you're using qmail (see IntegratedInMta), you can set up SpamAssassin through Qmail-Scanner, which has a patched version able to delete spam at a given threshold. Alternatively, you can configure the patched version of Qmail-Scanner to reject spam at a given threshold during the smtp session. Rejecting spam is a better approach than silently deleting or bouncing it because the sender (assuming its a real person) will know the message wasn't delivered and the sending mail server will have to handle the bounce which prevents back scatter from being sent by your mail server. Here's the patched version and the original.

If you don't want the full Qmail-Scanner you can try Mailparser by Eric Bambach. Originally a very lightweight C program but now (9/14/2007) re-implemented in perl it will drop any message that has X-Spam-Flag: YES in the header.

Rename /var/qmail/bin/qmail-queue to qmail-queue.orig and add it to the delivery queue after SpamAssassin.

Example /var/qmail/bin/qmail-queue file

```
#!/bin/bash
/usr/bin/spamc | /var/qmail/bin/mailparser | /var/qmail/bin/qmail-queue.orig
```

Challenge-response and Email passwords

If you're willing to make senders jump through hoops to reach you, you can begin rejecting all mails except for ones known to be legitimate. Some believe this solution is worse than the disease.

- Principles of a Challenge-Response system by Brad Templeton
- TMDA is the best known Challenge-Response system
- Email passwords by David Wheeler are another implementation

Contributors

- DanKohn
- David Wheeler