# Mail filtering examples



These are snippets of configuration for an Exim-based MTA. They have had some simple testing but are intended mainly as a starting point for building your own customised configurations. Test them for yourself, and read the related parts of the documentation.

---

## 1. RBL blacklists with opt-in/opt-out

Here we will configure Exim to check incoming mail against DNS RBLs, but individual users can have a different set of RBLs which apply to their account (which can be empty, in which case they have opted out of filtering altogether).

Customised RBLs can help improve the accuracy of your filtering. For example, if a particular user knows that they will never receive any legitimate mail from a particular country, then they can use an RBL which lists all IP addresses in that country (see http://blackholes.us/), without affecting any other users on your system.

Firstly, create a file `/usr/exim/dnslists` containing E-mail addresses (one per line), followed by a space or tab and the list of RBLs to use for that user. If you want a default policy to apply to all other users, put a line starting "*". For a policy which applies to all users in at domain.com, use "*@domain.com". If someone wants no RBL filtering at all, leave the right-hand side blank.

```
fred@flintstone.org      korea.blackholes.us : china.blackholes.us
wilma@flintstone.org
*                        sbl.spamhaus.org : ordb.relay.org : bl.spamcop.net
```

Now edit the configuration file `/usr/exim/configure`. In the section 'begin acl' locate the following lines:

```
  # deny    message       = rejected because $sender_host_address is in a black list at $dnslist_dor
  #         dnslists      = black.list.example
```

Uncomment these two lines and change the second line as shown here:

```
  deny    message       = rejected because $sender_host_address is in a black list at $dnslist_doma
          dnslists      = ${lookup{${lc:$local_part@$domain}}lsearch*@{/usr/exim/dnslists}}
```

How does this work? Instead of having a static list of dnslists, which is the same for everyone, we perform a file lookup to decide which dnslists to use for this particular recipient. This is using Exim's "string expansion" facility.

${lookup{key}lsearch*@{file}}
: Lookup the value "key" in the file "file". The result of the expansion is the rest of the line in the file.

lsearch*@
: We perform a linear search (top to bottom) through a plain text file. If the key "foo@bar" is not found, then we look a second time for "*@bar", and if still not found, look a third time for "*".

${lc:some-string}
: Expands "some-string" and converts it to Lower Case. This is because incoming mail might be using uppercase characters for some or all of the address, so we need to convert to all lower-case to find the key.

$local_part@$domain
: The E-mail address which we are currently processing

You can test and debug string expansions like this using exim's `-be` (expression testing) mode.

```
# /usr/exim/bin/exim -be '${lookup{fred@flintstone.org}lsearch*@{/usr/exim/dnslists}}'
korea.blackholes.us : china.blackholes.us
```

To test whether the policy works use exim's -bh mode which simulates an SMTP connection from a particular IP address. The address 61.32.0.1 is included in the korea.blackholes.us list, so:

```
# /usr/exim/bin/exim -bh 61.32.0.1
```

```
**** SMTP testing session as if from host 61.32.0.1
**** This is not for real!
220 noc.t1.ws.afnog.org ESMTP Exim 4.34 Wed, 19 May 2004 15:18:30 +0000
mail from:<>
250 OK
rcpt to:<wilma@flintstone.org>
250 Accepted
rcpt to:<fred@flintstone.org>
550-rejected because 61.32.0.1 is in a black list at korea.blackholes.us
550 Korea blocked by korea.blackholes.us
quit
221 noc.t1.ws.afnog.org closing connection
```

If the `/usr/exim/dnslist` file gets big, then it will be slow to search. In this case, you can convert it into an indexed .db file:

```
# /usr/exim/bin/exim_dbmbuild /usr/exim/dnslists /usr/exim/dnslists.db
```

You'll have to run this command every time you change dnslists. Then make another change to the configure file:

```
change          ....lsearch*@{/usr/exim/dnslists}}
to              ....dbm*@{/usr/exim/dnslists.db}}
```

For more information read the Exim manual, which is `doc/spec.txt` inside the source directory, or online at www.exim.org

# 2. Content filtering: exiscan-acl

Exim has some hooks which allow other people to write extensions which perform content scanning. One of these is "exiscan-acl". You need to apply the exiscan-acl patch before you compile exim. Assuming you have already downloaded and unpacked the exim source in /usr/exim:

```
$ cd /usr/exim
$ ftp noc.ws.afnog.org

Log in as 'anonymous'
ftp> cd /pub/t1
ftp> get exiscan-acl-4.34-21.patch
ftp> bye

$ cd /usr/exim/exim-4.34
$ patch -p1 <../exiscan-acl-4.34-21.patch
```

Now continue to build exim as instructed before (create and edit Local/Makefile and Local/eximon.conf). If you have built exim already, *before* you applied the exiscan-acl patch, then clean out the source tree by

```
$ rm -rf build-FreeBSD-i386

before continuing as before:
... set up Local/Makefile and Local/eximon.conf ...
$ make
$ su
# make install

Check the exim binary now includes exiscan:
# /usr/exim/bin/exim -bV
Exim version 4.34 #1 built 19-May-2004 15:47:58
...
Contains exiscan-acl patch revision 21 (c) Tom Kistner [http://duncanthrax.net/exiscan/]
```

Exiscan needs options in the configure file to enable it; until this is done, exim will continue to work just as it did before. To do useful filtering you will need to install SpamAssassin and/or clamav first. The Exiscan documentation is in **doc/exiscan-acl-spec.txt** and **doc/exiscan-acl-examples.txt** in the source directory once you've applied the exiscan patch.

# 3. SpamAssassin

```
Fetch Mail-SpamAssassin-2.63.tar.bz2 via ftp into /usr/exim
$ tar -xvyf Mail-SpamAssassin-2.63.tar.bz2
$ cd Mail-SpamAssassin-2.63
$ perl Makefile.PL
$ make
```

```
$ su
# make install
```

In the Mail-SpamAssassin-2.63 directory, files INSTALL and USAGE give more detailled information.

According to these instructions, you also need to install the package p5-HTML-Parser (also p5-HTML-Target which it depends on). It's easiest to do this using the packages system.

Now you need to create the configuration file, `/etc/mail/spamassassin/local.cf`. The easiest way to do this is using the web-based config generator at http://www.yrex.com/spam/spamconfig.php. Select "Disable Bayes System" (because we don't have a way to allow users to individually tag messages as spam) and "Disable RBL checks" (because Exim is already doing that for us, see above).

Now start the spamassassin daemon (this should go in /etc/rc.local as well):

```
# /usr/local/bin/spamd -d
```

You can test it manually using "spamc", a client program which sends mail to spamd for analysis:

```
$ spamc -R
Subject: penis enlagement

Great new pills available!!!!!!!
Ctrl-D
Content analysis details:   (3.5 points, 5.0 required)

 pts rule name              description
---- ---------------------- --------------------------------------------------
 1.9 FROM_NO_LOWER          'From' has no lower-case characters
 1.0 DATE_MISSING           Missing Date: header
 0.6 PENIS_ENLARGE2         BODY: Information on getting larger penis/breasts
```

Finally, you need to configure exiscan-acl to pass each message to spamd as it is received during the DATA phase, and reject if it's spam. The following additions to /usr/exim/configure come from doc/exiscan-acl-examples.txt as a starting point.

```
Add the following line in the top section of the file; a good place would
be next to the existing extry for acl_check_rcpt

acl_smtp_data = acl_check_data

Add the following in section 'begin acl', either before or after the
existing entry for smtp_check_rcpt

# This ACL is used for every DATA command in an incoming SMTP connection
acl_check_data:

accept hosts = 127.0.0.1 : +relay_from_hosts

deny message = $found_extension files are not accepted here
     demime = com:vbs:bat:pif:scr

deny message = Serious MIME defect detected ($demime_reason)
     demime = *
     condition = ${if >{$demime_errorlevel}{2}{1}{0}}

# Comment out virus scanner until you have configured it
# deny message = This message contains malware ($malware_name)
#      demime = *
#      malware = *

deny message = Classified as spam (score $spam_score)
     condition = ${if <{$message_size}{80k}{1}{0}}
     spam = nobody

accept
```

Now test:

```
# /usr/exim/bin/exim -bh 1.2.3.4
mail from:<>
250 Accepted
rcpt to:<fred@flintstone.org>
250 Accepted
data
354 Enter message, ending with "." on a line by itself
Subject: penis enlargement

make lots of money fast!!!!!!!
.
550 Classified as spam (score 3.5)
```

If you want to set a lower spam score than the default (5.0) for testing, then remember to kill and restart spamd after editing /etc/mail/spamassassin/local.cf

---

# 4. ClamAV

The documentation is on-line at [http://clamav.sourceforge.net/doc/](http://clamav.sourceforge.net/doc/) and the important thing to note is you need a 'clamav' user and group before starting to compile, and the 'clamav' user must be in the 'exim' group so that it can access spool files.

```
$ su
# pw useradd clamav -s /bin/false -c "Clam AntiVirus"
# pw usermod clamav -G exim
# exit

Fetch clamav-0.71.tar.gz via ftp into /usr/exim
$ tar -xvzf clamav-0.71.tar.gz
$ cd clamav-0.71
$ ./configure
$ make
$ su
# make install
# vi /usr/local/etc/clamav.conf

Comment out the 'Example' line
Uncomment 'PidFile /var/run/clamd.pid'
Uncomment 'Temporary Directory /var/tmp'
Uncomment 'User clamav'
Uncomment 'AllowSupplementaryGroups'
Uncomment 'ScanMail'

Start the daemon
# /usr/local/sbin/clamd        # goes in /etc/rc.local as well
# exit

Now run the virus scanning tests; make sure you are still inside
the clamav-0.71 source directory
$ pwd
/home/inst/clamav-0.71
$ clamdscan -l scan.txt .
...
----------- SCAN SUMMARY -----------
Infected files: 6
Time: 4.051 sec (0 m 4 s)

The scanning results are shown on the screen and are stored in scan.txt
```

Now to configure exim:

```
# vi /usr/exim/configure

Add this line somewhere in the top section
av_scanner = clamd:/tmp/clamd

Using the spamassassin ACL from earlier, uncomment these lines
which were previously commented out

 deny message = This message contains malware ($malware_name)
      demime = *
      malware = *
```

To test, clamav comes with test virus signatures. One is a single line of text that you can copy-paste into an E-mail message, other are attachments. The exact signature is not reproduced here so that this file does not appear to contain a virus!

```
$ cat test/test1
$CEliacma........

# /usr/exim/bin/exim -bh 1.2.3.4
220 noc.t1.ws.afnog.org ESMTP Exim 4.34 Thu, 20 May 2004 10:33:17 +0000
mail from:<>
250 Accepted
rcpt to:<fred@flintstone.org>
250 Accepted
data
354 Enter message, ending with "." on a line by itself
Subject: test

$CEliacma........
.
550 This message contains malware (ClamAV-Test-Signature)
```