Exercises: SpamAssassin Install with Exim: PacNOG I Workshop

June 20, 2005

Note: The "#" and "$" characters before commands represents your system prompt and is not part of the command itself. "#" indicates a command issued as root while "$" indicates a command issued as a normal user.

Note 2: If you install software, update your environment as root and the change is not immediately available try typing `rehash` at the root shell prompt. This is only necessary when running a C shell (e.g., like /bin/csh).

Note 3: These exercises are based on materials from Philip Hazel.

Basic SpamAssassin Installation using Ports

You need to be root to do this. Using ports downloads all the dependencies, and there are a lot of them!

```
# cd /usr/ports/mail/p5-Mail-SpamAssassin
# make install
```

This may take a while to complete. In the directory `/usr/local/share/doc/p5-Mail-SpamAssassin` the files `INSTALL` and `USAGE` give more detailed information. In addition the web site http://spamassassin.apache.org/ has additional documentation and software.

Now setup the SpamAssassin configuration to disable the most expensive network and cpu-based tests.

```
# cd /usr/local/etc/mail/spamassassin
# cp local.cf.sample local.cf
# vi local.cf
```

Add the following lines to the end of the file:

```
use_dcc 0
use_pyzor 0
use_razor2 0
skip_rbl_checks 1
use_bayes 0
```

Now you need to configure `/etc/rc.conf` so that the SpamAssassin daemon will start automatically each time your machine boots, and so that you can execute the SpamAssassin start script (remember you need this entry, otherwise the script won’t work).

First, add the following line to `/etc/rc.conf`:

```
spamd_enable="YES"
```

Now start the SpamAssassin daemon, which is called *spamd*:

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

Check that the SpamAssassin daemon is running. You may see multiple instances of the daemon:

```
# ps auxw | grep spamd
```

You can test the *spamd* daemon manually using *spamc*, a client that sends mail to *spamd* for analysis. You can do this as a regular user:

```
$ spamc -R

subject: penis enlargement
```

```
Great new pills available!!!!!
Ctrl-D
```

The output should look something like this:

```
-2.0/5.0
Spam detection software, running on the system "localhost.", has
identified this incoming email as possible spam.  The original message
has been attached to this so you can view it (if it isn't spam) or label
similar future email.  If you have any questions, see
the administrator of that system for details.

Content preview:  Great new pills available!!!! [...]

Content analysis details:   (-2.0 points, 5.0 required)

 pts rule name              description
---- ---------------------- --------------------------------------------------
 0.0 MISSING_DATE           Missing Date: header
-2.8 ALL_TRUSTED            Did not pass through any untrusted hosts
 0.8 BODY_ENHANCEMENT2      BODY: Information on getting larger body parts
```

Despite its content this message has been strongly tagged as 'not spam' because it has not been through any 'untrusted' hosts. You may not see exactly this output: it depends on the default SpamAssassin configuration that has been installed.

We are not going to change the Exim configuration so that every message is passed to SpamAssassin. At first, we won't block any messages. Instead, we will put the spam score and other SpamAssassin output into new headers that are added to the message:

Edit the file */usr/local/etc/exim/configure* and find the line that contains:

```
acl_smtp_rcpt - acl_check_rcpt
```

Remember, you can use "/string" in vi to find this line quickly.

Add the following new lines to the file just below this line:

```
acl_smtp_data = acl_check_data
acl_not_smtp = acl_check_data
```

The first line asks Exim to run an ACL check when a message's data has been received in an SMTP transaction. The second line asks for the same ACL to be run on non-SMTP messages. This ensures that all incoming messages are scanned. We must now define the ACL.

Find the configuration line that contains:

```
begin acl
```

And, insert the following lines directly below that line:

```
acl_check_data:
  warn  spam    = nobody
        message = X-is-spam: over spam threshold
  warn  message = X-Spam_score: $spam_score\n\
                  X-Spam_score_int: $spam_score_int\n\
                  X-Spam_bar: $spam_bar\n\
                  X-Spam_report: $spam_report
  accept
```

The **warn** verb in an ACL doesn't accept or reject, but if its conidtions are true, it can add headers to the message. The first **warn** passes the message to SpamAssassin, and if the spam score is over the threshold, an `X-is-spam:` header is added. The second **warn** adds some more headers containing information from SpamAssassin. Theare are always added, unconditionally.

Now send yourself a spam-like message. This examples uses the local SMTP interface (you can do this with a regular account):

```
$ exim -bs

mail from:<>
rcpt to:<username@yourhostname>
data
message-id: abcd
subject: BUY VIAGRA HERE!!!

<html><p>Dear Friend</p>
<p>VIAGRA $10.00</p>
<p>RISK FREE</p></html>
.
quit
```

Take a look in your mailbox and at the headers of the message that you receive.

Change the first **warn** in the ACL definition above to **deny**, and try the test again. If the spam score is at least 5, the message should be rejected.

Hervey Allen
Philip Hazel

Last modified: Tue Jun 14 03:08:34 CLT 2005