Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

Control antispam con Postfix+SpamAssassin (50510 lectures)

Per **Josep Sort**, ***Josep*** (http://infoterrassa.com)

Creado el 22/06/2003 19:06 modificado el 22/06/2003 19:09

> *Hace poco decidí abandonar sendmail para implementar Postfix según el artículo Cómo montar un potente sistema de correo con Postfix[1]. El problema que tiene el servidor de correo montado de esta forma (con dominios y usuarios virtuales) es que no se pueden procesar los mensajes mediante procmail, lo cual me impedía controlar la gran plaga de Internet: el spam. Mediante algunos trucos, conseguí realizar este control. La respuesta: amavisd-new*

Hasta ahora, el filtrado de los mensajes lo llevaba a cabo mediante *procmail*, pero debido a la naturaleza de esta aplicación, no es posible procesar los mensajes cuando estos van destinados a usuarios y dominios virtuales. Ello implica que no pueda aplicar el único filtro que realmente necesitaba: en control antispam. Buscando una solución, encontré un mensaje[2] del Kansas City LUG[3] en el cual se dan algunas pistas para llevar a cabo este control mediante amavisd-new y SpamAssassin.

Instalar amavisd-new y SpamAssassin

En Debian (lo siento, yo también he caido...):
apt-get install spamassassin
apt-get install amavisd-new

---

amavisd-new

Esta versión de *AMaViS (A Mail Virus Scanner)* se diferencia de la anterior en que recoge el correo desde un puerto del servidor de correo y lo devuelve a otro puerto, además de que *sabe usar* SpamAssassin. Lo que hay que conseguir es que Postfix envíe el correo a este puerto antes de hacer la entrega. Luego, *amavisd-new* devuelve el mensaje a Postfix y este hace la entrega definitiva.

- Modificar */etc/postfix/master.cf* y añadir, al final:

```
 amavis-smtp[tab]unix[tab]-[tab]-[tab]y[tab]-[tab]2[tab]smtp
  -o smtp_data_done_timeout=1200s
  -o smtp_never_send_ehlo=yes
  -o disable_dns_lookups=yes
localhost:10025[tab]inet[tab]n[tab]-[tab]y[tab]-[tab]-[tab]smtpd
  -o content_filter=
  -o local_recipient_maps=
  -o smtpd_helo_restrictions=
  -o smtpd_client_restrictions=
  -o smtpd_sender_restrictions=
  -o smtpd_recipient_restrictions=permit_mynetworks,reject
  -o mynetworks=127.0.0.0/8
```

Aviso: *No debe haber espacios al final de cada línea. Allí donde se indica [tab] se debe cambiar por* ***un*** *tabulador.*

- Modificar */etc/postfix/main.cf* y añadir:

```
 content_filter = amavis-smtp:localhost:10024
```

- Modificar */etc/amavis/amavisd.conf*

Los cambios en este fichero de configuración son bastante triviales y se basa casi exclusivamente en descomentar algunas líneas, comentar otras y algún que otro cambio. Para el que quiera tener el control de la situación y ajustarlo a sus necesidades, sólo algunas indicaciones:

- La dirección de correo del administrador (de spams o de virus) debe ser completa, o sea, con el dominio incluido. Esto es porque tenemos Postfix configurado para usuarios y dominios virtuales. La arroba de la dirección se debe indicar como \@ (p.e. spam\@dominio.com).
- Si no se usa análisis vírico de los mensajes, se deben comentar todas las líneas en las cuales hace referencia.
- Aquí se asume que se ha creado una cuenta llamada *spam* que será la que recogerá todos los mensajes clasificados como tal.

Y para el más práctico, aquí[4] hay el mío. Se debe cambiar "dominio.com" por el dominio correcto. En vim:

```
:%s/dominio.com/minombrededominio.org/g
ZZ
```

Este archivo de configuración envía todos los mensajes de *spam* a una única cuenta de correo. Los usuarios no recibirán estos mensajes en ningún momento. Para que sea el usuario el que controle sus *spams* este archivo no sirve y hay que currárselo a mano[5].

SpamAssassin

Esta aplicación realiza una serie de pruebas[6] a los mensajes. Para cada prueba que supera, le asigna una puntuación. Cuando la puntuación llega a 5 (valor por defecto) entiende que se trata de un mensaje de spam. Además, desde hace poco SpamAssassin incluye también filtros *bayesianos* que permiten ajustar delicada y automáticamente la clasificación de un mensaje. Así, en caso de un falso positivo/negativo, podemos adiestrar a SpamAssassin para que tenga en cuenta la característica de éste en el futuro. Los filtros *bayesianos* necesitan un gran número de mensajes para *aprender*, por lo cual deberemos estar una buena temporada *enseñándole*.
Los filtros *bayesianos* están activados por defecto, y SpamAssassin *adiestra* este filtro según su tabla de reglas. Si un mensaje supera las pruebas y es clasificado como spam, añade las reglas de este mensaje a la base de datos *bayesiana*. Lo mismo si un mensaje no lo clasifica como spam. Así, si las reglas propias funcionan correctamente en vuestro caso esta base de datos se alimentará sin la atención del usuario. Esto sería en un mundo ideal, en el País de las Maravillas, pero esto no es así, desgraciadamente, en la realidad. Por lo tanto debemos preparar un sistema que nos facilite este control.
Los pasos serán:

- Crear la cuenta de spam, que deberá ser de tipo *imap*. Si el servidor de correo es Postfix+Mysql[1], las cuentas ya serán de este tipo.
- Desde el cliente de correo (KMail o el que sea), crearemos una nueva cuenta *imap* llamada *Spam* y que accederá a la cuenta *spam* del servidor de correo.
- Mediante el cliente de correo, crearemos dos subcarpetas de la carpeta "Entrada" de la cuenta *Spam*: una llamada *Spam-Si* y otra llamada *Spam-No*
- Periódicamente, el servidor de correo recogerá los mensajes del directorio */var/mail/midominio.net/spam/.Spam-No/cur* y de */var/mail/midominio.net/spam/.Spam-Si/cur* y readiestraremos SpamAssassin según la nueva regla.

De esta forma, si recibimos un falso positivo (mensaje legítimo clasificado erróneamente como *spam*) sólo lo arrastraremos a la subscarpeta "Spam-No". Y al revés, en caso de falso negativo (mensaje de spam entregado como legítimo), lo arrastramos a la carpeta "Spam-Si". No hará falta borrarlos, puesto que será el mismo servidor el que lo haga. Los falsos positivos, no los moveremos, sino que los *copiaremos* (arrastrándolos pulsando la tecla Control) a la carpeta "Spam-No". El original, lo guardaremos en una carpeta legítima, a buen recaudo.

Para ello:

- Entramos al servidor de correo como "root" y editamos su crontab (crontab -e)

```
3 * * * * /bin/mv /var/mail/midominio.net/spam/.Spam-Si/cur/* /var/lib/amavis/spam/. >/dev/null 2>/dev/null
3 * * * * /bin/mv /var/mail/midominio.net/spam/.Spam-No/cur/* /var/lib/amavis/ham/. >/dev/null 2>/dev/null
4 * * * * /bin/chown amavis.amavis /var/lib/amavis/spam/* >/dev/null 2>/dev/null
4 * * * * /bin/chown amavis.amavis /var/lib/amavis/ham/* >/dev/null 2>/dev/null
```

- Una vez dentro, hacemos "su amavis" para actuar como el usuario "amavis" que es el responsable de procesar el correo.
- Escribimos "cd" sin parámetros para situarnos en el directorio de inicio de AMaVis.
- Creamos dos directorios, uno llamado *spam* (para procesar los mensajes de spam) y otro llamado *ham* para procesar los mensajes legítimos.
- Editamos el crontab (crontab -e) y añadimos las siguientes órdenes:

```
5 * * * * /usr/bin/sa-learn --forget --dir /var/lib/amavis/spam >/dev/null 2>/dev/null
5 * * * * /usr/bin/sa-learn --forget --dir /var/lib/amavis/ham >/dev/null 2>/dev/null
6 * * * * /usr/bin/sa-learn --spam --dir /var/lib/amavis/spam >/dev/null 2>/dev/null
6 * * * * /usr/bin/sa-learn --ham --dir /var/lib/amavis/ham >/dev/null 2>/dev/null
7 * * * * /bin/rm -f /var/lib/amavis/spam/* >/dev/null 2>/dev/null
7 * * * * /bin/rm -f /var/lib/amavis/ham/* >/dev/null 2>/dev/null
```

Qué hace esto? Yo tengo la costumbre de recoger el correo externo mediante *fetchmail* cada media hora. Entonces, a la horas y tres minutos, muevo el contenido de las subcarpetas *imap* "Spam-Si" y "Spam-No" a los directorios *spam* y *ham*, respectivamente, del usuario "amavis". Se debe hacer como "root" debido a los permisos restrictivos de los directorios que estamos tratando y también porque estamos moviendo información entre usuarios distintos. Una vez movidos los mensajes, los cambiamos de propietario.
El crontab del usuario "amavis" adiestrará los filtros bayesianos de la siguiente forma:

- Puesto que SpamAssassin *aprende* automáticamente, lo primero que debemos decirle es que "olvide" lo que ha aprendido de estos mensajes mediante la orden "sa-learn --forget". Debemos aplicarlo a los directorios *spam* y *ham*
- Adiestramos SpamAssassin con la información correcta.
  - "sa-learn --spam" *aprende* del contenido del directorio *spam* (--dir spam) de que esos mensajes son spam
  - Por otro lado, "sa-learn --ham" *aprende* del contenido del directorio *ham* (--dir ham) de que esos mensajes **no** son spam
- Una vez terminado, borramos el contenido de los directorios *spam* y *ham* y los dejamos preparados para la próxima misión.

Los mensajes que se hayan clasificado correctamente como *spam* simplemente los borramos.
Y bueno, eso es todo. ¡Sed felices "asesinando" spams!

Lista de enlaces de este artículo:

1. http://bulma.net/body.phtml?nIdNoticia=1621
2. http://www.kclug.org/archives/2002/oct/0493.shtml
3. http://www.kclug.org
4. http://infoterrassa.com/stuff/amavisd.conf.gz
5. http://www.avecrem.com
6. http://www.spamassassin.org/tests.html

E-mail del autor: josep _ARROBA_ sortnet.com
Podrás encontrar este artículo e información adicional en: http://bulma.net/body.phtml?nIdNoticia=1799