Uso de listas blancas y SPF en SpamAssassin

Autor: Pablo Fernández Baladrón (Servicios Informáticos de la Universidad de Vigo)

Se pueden utilizar listas blancas DNS en SpamAssassin declarándolas de forma similar a una lista negra y asignándoles una puntuación negativa muy alta. Esto hará que el mensaje se acepte siempre, independientemente de la puntuación asignada al mensaje por otras pruebas en SpamAssassin.

En las comprobaciones en que se utilizan consultas DNS como listas negras o blancas DNS o comprobación de registros SPF, SpamAssassin utiliza en principio todas las cabeceras “Received:” del mensaje procesado.

Esto, que para comprobar si un correo proviene de un servidor en listas negras puede tener sentido (se comprueban todos los servidores por los que el correo parece haber pasado), para comprobaciones en listas blancas no es deseable ya una cabecera “Received:” falsificada haría parecer que el correo proviene de un servidor de confianza y, por tanto, sería sencillo saltarse todos los demás controles antispam.

Hay algo que hay que tener en cuenta al hacer estas comprobaciones es que SpamAssassin necesita recibir el mensaje incluyendo la última cabecera “Received:” insertada, que es la que añade nuestro servidor de correo al recibir el mensaje. Esta cabecera incluye la dirección IP y nombre del servidor que nos envía el correo, que es lo que necesitamos para hacer estas comprobaciones y que, por otra parte, es la primera cabecera de este tipo cuyo contenido es de fiar (las otras podrían haber sido falsificadas).

Según la forma en que se comunique el servidor de correo con SpamAssassin incluirá la cabecera el propio servidor antes de enviarla a spamd o será necesario, en algunos casos (filtros milter para sendmail, por ejemplo), que el filtro añada una cabecera “Received:” falsa que incluya estos datos.

Uso de listas blancas ESWL / MTAWL

Para utilizar las listas blancas DNS es necesario que los chequeos de listas negras en SpamAssassin estén habilitados. Por defecto lo están, pero en caso de ser necesario se pueden habilitar con indicar (en Debian en /etc/spamassassin/local.cf):

```
skip_rbl_checks 0
```

Para declarar una lista blanca se emplea la función check_rbl de SpamAssassin, a la que se pasan como parámetros:

- Nombre identificativo utilizado para esta lista
  - Sólo tiene utilidad real cuando se tienen listas que permiten subconsultas (tipo combined.njabl.org, dnsbl.sorbs.org, sbl-xbl.spamhaus.org, etc.)
  - A este nombre identificativo se le pueden agregar sufijos que alteran la forma en que SpamAssassin hace los chequeos:
    - firsttrusted:
      - Sólo se aplica la comprobación a las cabeceras "Received:" agregadas por equipos de confianza. En una estafeta de correo de entrada sólo se comprobará la agregada por el propio servidor de correo.
    - notfirsthop:
      - No se aplica la comprobación a la cabecera Received más antigua. Se utiliza para evitar que se detecten como SPAM mensajes procedentes de una IP de ADSL o módem que envíe sus mensajes de forma legítima a través de un servidor de correo de su proveedor. No tiene mucho sentido utilizarlo para este tipo de comprobaciones.
- Dominio que se consultará
  - La función check_rbl de SpamAssassin hace las consultas invirtiendo la dirección IP, por ejemplo para *130.206.1.3* buscaría, en ESWL, el nombre *3.1.206.130.eswlrev.dnsbl.rediris.es*

La lista blanca se declarará como:

- ESWL:

```
header RCVD_IN_ESWL_WHITELIST    eval:check_rbl('eswlrev-firsttrusted',
'eswlrev.dnsbl.rediris.es')
describe RCVD_IN_ESWL_WHITELIST  Relay in eswl whitelist
tflags RCVD_IN_ESWL_WHITELIST    net
score RCVD_IN_ESWL_WHITELIST     -99
```

- MTAWL:

```
header RCVD_IN_MTAWL_WHITELIST    eval:check_rbl('mtawlrev-firsttrusted',
'mtawlrev.dnsbl.rediris.es')
describe RCVD_IN_MTAWL_WHITELIST  Relay in mtawl whitelist
tflags RCVD_IN_MTAWL_WHITELIST    net
score RCVD_IN_MTAWL_WHITELIST -99
```

Comprobación de registros SPF

El uso de SPF desde SpamAssassin es una alternativa razonable a filtros que utilicen las librerías libspf ó libspf2, ya que en algunas distribuciones Linux (caso de Debian Sarge) las versiones que se distribuyen tienen problemas de estabilidad (pérdidas de memoria, etc.).

El mayor problema que se tiene, al menos con las versiones probadas (SpamAssassin 3.0.3 con Mail::SPF::Query 1.997) es que se sólo se hace la verificación de registros SPF si el mensaje contiene la cabecera "Return-Path", por lo que resulta fácil saltarse esta comprobación.

En principio la comprobación que importaría sería SPF_FAIL, las demás se podrían poner con puntuación 0 (en principio no parece recomendable el utilizar las comprobación sobre el nombre de servidor indicado en el HELO/EHLO, SPF_HELO_FAIL).

Para utilizar las comprobaciones SPF es necesario previamente cargar el plugin de SPF (esto en Debian se hace en el fichero /etc/spamassassin/init.pre):

```
loadplugin Mail::SpamAssassin::Plugin::SPF
```

Una vez cargado el plugin de SPF, las comprobaciones de registros SPF se deberían tener en los ficheros de configuración por defecto de SpamAssassin (en Debian en /usr/share/spamassassin/25_spf.cf):

```
ifplugin Mail::SpamAssassin::Plugin::SPF
header SPF_PASS                 eval:check_for_spf_pass()
header SPF_FAIL                 eval:check_for_spf_fail()
header SPF_SOFTFAIL             eval:check_for_spf_softfail()
header SPF_HELO_PASS            eval:check_for_spf_helo_pass()
header SPF_HELO_FAIL            eval:check_for_spf_helo_fail()
header SPF_HELO_SOFTFAIL        eval:check_for_spf_helo_softfail()
describe SPF_PASS               SPF: sender matches SPF record
describe SPF_FAIL               SPF: sender does not match SPF record (fail)
describe SPF_SOFTFAIL           SPF: sender does not match SPF record (softfail)
describe SPF_HELO_PASS          SPF: HELO matches SPF record
describe SPF_HELO_FAIL          SPF: HELO does not match SPF record (fail)
describe SPF_HELO_SOFTFAIL      SPF: HELO does not match SPF record (softfail)
tflags SPF_PASS                 net nice userconf
tflags SPF_FAIL                 net
tflags SPF_SOFTFAIL             net
tflags SPF_HELO_PASS            net nice userconf
tflags SPF_HELO_FAIL            net
tflags SPF_HELO_SOFTFAIL        net
score SPF_PASS 0
score SPF_FAIL 8
score SPF_SOFTFAIL 0
score SPF_HELO_PASS 0
score SPF_HELO_FAIL 0
score SPF_HELO_SOFTFAIL 0
endif # Mail::SpamAssassin::Plugin::SPF
```

Comprobación del funcionamiento de ESWL / MTAWL

Se pueden comprobar las puntuaciones asignadas a un mensaje por SpamAssassin usando spamassassin con la opción "-t", de la forma:

```
spamassassin -t  < mensaje-a-comprobar
```

La salida de este comando mostrará el mensaje que se está utilizando para hacer la comprobación, añadiendo las cabeceras de SpamAssassin, la puntuación obtenida por el mensaje y las reglas que se aplicaron para obtener esa puntuación. Se puede añadir la opción "-D" para obtener información de depuración adicional (se envía esta información a STDERR).

Por ejemplo para el siguiente mensaje:

```
Return-Path: <pablof@uvigo.es>
Received: from fire.cesga.es (fire.cesga.es [193.144.34.133])
        by mail.uvigo.es (8.13.4/8.13.4/Debian-3sarge2) with ESMTP id k8P8uho703
1594
        for <pablof@uvigo.es>; Mon, 25 Sep 2006 10:56:45 +0200
Received: from cosa (localhost [127.0.0.1])
        by fire.cesga.es (8.12.9/8.12.9) with SMTP id k8P8uULQ003951
        for pablof@uvigo.es; Mon, 25 Sep 2006 10:56:38 +0200 (CEST)
Date: Mon, 25 Sep 2006 10:56:30 +0200 (CEST)
Message-Id: <200609250856.k8P8uULQ003951@fire.cesga.es>
Subject: Correo de prueba
From: Pablo <pablof@uvigo.es>
To: <pablof@uvigo.es>
X-Greylist: Sender IP whitelisted, not delayed by milter-greylist-2.0.2 (mail.uv
igo.es [193.146.32.69]); Mon, 25 Sep 2006 10:56:45 +0200 (CEST)
X-Mail-Scanned: Criba 2.0 + Clamd & Spamassassin

A ver....
```

Se tendrá como salida:

```
spamassassin -t -D < mensaje-de-prueba
...
[texto del mensaje original]
[texto del mensaje con las cabeceras de SpamAssassin añadidas]
...
Content analysis details:   (16.0 points, 7.0 required)

 pts rule name              description
---- ---------------------- --------------------------------------------------
 8.0 SPF_FAIL               SPF: sender does not match SPF record (fail)
[SPF failed: Please see
http://spf.pobox.com/why.html?sender=pablof%40uvigo.es&ip=193.144.34.133&receive
r=acelga]
 8.0 BAYES_99               BODY: Bayesian spam probability is 99 to 100%
                            [score: 1.0000]
```