Controles de acceso en Postfix

En la gestión de un servidor de correo hoy día, cada vez es más importante el tema de la seguridad, en distintos aspectos. En este documento nos limitaremos a los mecanismos de Postfix para controlar en qué condiciones de acepta un mensaje para su procesamiento (entrega local, a otro servidor, etc.).

Un mensaje de correo-e puede entrar en Postfix de forma local (mediante el programa `/usr/lib/sendmail`, que presenta una interfaz compatible en gran medida con el mismo programa de Sendmail) o mediante una conexión SMTP. En el primer caso, el mensaje se acepta siempre. Es en el segundo caso cuando entran en acción los controles de acceso (llamados **UCE controls** en la documentación de Postfix). Sin embargo aún tenemos mecanismos para rechazar mensajes tanto si llegan de una manera como de otra: el filtrado (content_filter) y las comprobaciones de cabeceras (header_checks) y de contenido (body_checks). Como hemos dicho, nos centraremos sólo en los controles de acceso por SMTP y no en estos otros mecanismos. Tampoco entraremos en temas de autentificación o comunicaciones seguras entre servidores de correo (SASL, TLS).

En Postfix, por defecto, un mensaje que llega por SMTP se acepta sólo si se cumple alguna de las siguientes condiciones:

- El cliente tiene una dirección IP que pertenezca al valor del parámetro `mynetworks` (por defecto es nuestra subred).
- El nombre de host del cliente (previa consulta inversa a DNS) pertenece a alguno de los dominios incluidos en el parámetro `relay_domains` (por defecto su valor es el de `mydestination`, cuyo valor por defecto a su vez es el de nuestro nombre de host) o a subdominios suyos.
- La dirección de destino (RCPT TO) pertenece a alguno de los dominios incluidos en el parámetro `relay_domains` o a subdominios suyos, sin importar en este caso la dirección del cliente.

Si usamos direcciones y/o dominios virtuales, la cosa se complica un poco, pero no demasiado, como veremos.

Éste es el comportamiento por defecto, sin embargo tenemos a nuestra disposición mecanismos muy flexibles para adaptarlo a nuestras necesidades.

Existen cuatro parámetros de configuración principales para controlar el acceso, cada uno de los cuales puede tomar como su valor una serie de restricciones. Postfix evalua la serie de restricciones que componen cada uno de estos parámetros tras cada una de las primeras fases del diálogo SMTP con el cliente. Las fases y los parámetros son:

Conexión	`smtpd_client_restrictions`
HELO/EHLO	`smtpd_helo_restrictions`
MAIL FROM	`smtpd_sender_restrictions`
RCPT TO	`smtpd_recipient_restrictions`

A partir de ahora, a estos parámetros los llamaremos RC (del término usado en la documentación, Restriction Class). Las restricciones que pueden usarse se detallan en la tabla que se muestra más abajo. Cada restricción puede devolver uno de los siguientes tres valores posibles: Aceptar, Rechazar, No-Sé. Por ejemplo, una restricción puede ser buscar la dirección IP del cliente en un fichero con el siguiente formato:

```
a.b.c.d OK
e.f.g.h REJECT
i.j.k.l 450 Fallo temporal
```

Si la dirección IP del cliente es `a.b.c.d`, el mensaje será aceptado. Si aquélla es `e.f.g.h`, será rechazado. Si es `i.j.k.l`, será rechazado, pero con un fallo temporal, con lo que el cliente volverá a intentar su entrega más tarde. Si la dirección IP del cliente es cualquier otra, el resultado de la restricción será No-Sé, con lo cual se evaluará la siguiente restricción, si la hay.

Cuando Postfix evalúa, tras cada fase del diálogo SMTP, la lista de restricciones que componen el parámetro `smtpd_xxx_restrictions` correspondiente a esa fase, actúa así:

- En cuanto una restricción devuelve algo que implica aceptar el mensaje, no se evalúan las restricciones que quedan y el resultado neto de la evaluación es aceptar.
- Lo mismo se puede decir en caso de rechazo.
- En caso de que una restricción devuelva No-Sé, se evalúan las siguientes restricciones. Si se llega al final, el resultado neto es No-Sé, que en este caso significa aceptar.

Si no se desea que el No-Sé final implique aceptación, se puede poner `reject` como restricción final.

Como caso especial, por seguridad, el parámetro `smtpd_recipient_restrictions` (más adelante veremos por qué esta limitación se pone precisamente en este parámetro) debe incluir al menos una de las siguientes restricciones que nunca pueden devolver No-Sé: `reject_unauth_destination`, `check_relay_domains`, `reject`.

La evaluación de las restricciones se efectúa de forma secuencial, tal como aparecen en el fichero de configuración (`main.cf`).

Resultado de la evaluación de las RCs

Éste es un punto que no está bien explicado en la documentación de Postfix y confunde a muchos administradores.

Si una RC evalúa a rechazar el mensaje, éste es rechazado, sin que haga falta evaluar las RCs de las fases posteriores. Pero si evalúa a aceptar o No-Sé, sí se evalúan dichas RCs. Como el resultado por defecto de las tres primeras RCs en Postfix es No-Sé (de hecho, por defecto no tienen ningún valor asignado), no tiene sentido incluir, en esas tres RCs, restricciones que no puedan devolver como resultado un rechazo. O sea, si queremos rechazar conexiones por dirección/hostname del cliente (caso de un bombardeo o un indeseable, por ejemplo, o permitir conexiones sólo de una serie de máquinas internas) o por nombre de host declarado en HELO/EHLO o por dirección de correo origen (casos estos poco frecuentes por lo fácil de falsear la información proporcionada) entonces sí tiene sentido poner restricciones en las tres primeras RCs. En caso contrario no lo tiene. Ésa es la razón por la que mucha gente pone todas las restricciones en la última RC.

La idea básica es que hasta que no se pasa la fase `RCPT TO`, normalmente no se puede saber si el mensaje debe ser aceptado o rechazado, pues sea quien sea la máquina cliente y la dirección origen, si el destinatario es local normalmente se aceptará.

(existen casos donde no es así; por ejemplo si obligamos a que si una conexión viene de una máquina interna la dirección origen sea del tipo `...@midominio.com`, esto se podría detectar en la fase `MAIL FROM`; este tipo de controles se pueden hacer pero de forma más compleja).

Por tanto, la idea es no rechazar nada antes de que pase esa fase, salvo que tengamos razones concretas para ello.

Pongamos un ejemplo: queremos tener los controles habituales, pero también que una serie de usuarios amigos de otros dominios puedan usar nuestro servidor para enviar correo a donde quieran. Lo que puede parecer más lógico es:

`main.cf`

```
smtpd_client_restrictions = check_client_access hash:/etc/postfix/amigos
```

`/etc/postfix/amigos`

```
host.friend1.com OK
host.friend2.org OK
```

Esto no tiene ningún efecto, porque en el caso de conexiones de estos hosts, el resultado de la primera RC será Aceptar, en vez del habitual No-Sé, y ya hemos visto que ambos tiene el mismo significado como resultado final de un RC. Si desde una de esas máquinas se quiere enviar un mail a una dirección que no sea de nuestro dominio, la cuarta RC lo rechazará. Para conseguir lo que queremos habría que hacer es, usando el mismo fichero `amigos`, poner en `main.cf`:

```
smtpd_recipient_restrictions = permit_mynetworks,
                               check_client_access hash:/etc/postfix/amigos,
                               check_relay_domains
```

De esta manera, si el resultado de esa restricción es Aceptar, éste será asimismo el de toda la RC, y como es la última, se aceptará.

RCs definidas por el administrador

Se pueden definir nuevas RCs, cuya utilidad principal es poder usarlas en la parte derecha de las tablas de acceso, donde por cierto también se pueden usar los nombres de las RCs que hemos visto hasta ahora, y que para no confundirnos llamaremos RCs standard. La idea es que en una entrada de una tabla de acceso, no se devuelva directamente aceptación o rechazo, sino que se invoque a un RC cuyo valor final será el de dicha entrada.

Una aplicación interesante de este mecanismo es definir reglas de acceso en base a más de un parámetro. Las

restricciones en Postfix (ver más adelante en este mismo documento) se evaluan siempre sobre un parametro (excepto `check_relay_domains` y algunas más), como las direcciones de origen o destino, dirección IP del cliente, etc. Usando RCs podemos saltarnos esta restricción. La mejor forma de explicarlo es un ejemplo: Queremos que los mails enviados desde nuestra red solo puedan llevar como remitente direcciones (en MAIL FROM) del tipo ...*@midominio.com*. Si no es así, serán rechazadas por el servidor. Vemos que la decisión depende de la relación entre dos datos: la dirección IP del cliente y la dirección de correo-e del remitente. Veamos cómo implementarlo:

main.cf

```
smtpd_restriction_classes = misremitentes
misremitentes = check_sender_access regexp:/etc/postfix/misremitentes
smtpd_sender_restrictions = check_client_access hash:/etc/postfix/misclientes
```

/etc/postfix/misremitentes

```
/@(.*\.)?midominio\.com$/ OK
/.*/ 554 La direccion remitente debe ser local
```

/etc/postfix/misclientes

```
midominio.com misremitentes
```

La comprobación se puede hacer en este caso tras la fase `MAIL FROM`, y como supone un comportamiento más restrictivo que el inicial de Postfix, se puede poner en `smtpd_sender_restrictions` en vez de en la última RC, como dijimos anteriormente. En ese momento se comprueba primero el nombre del host cliente (`check_client_access` usa la dirección IP del cliente y su nombre de host -obtenido por consulta DNS inversa- en la tabla que le demos) en la tabla `misclientes`. Si la máquina no pertenece a nuestro dominio, el resultado será No-Sé, o sea, se le permite pasar a la siguiente fase. Si es de nuestro dominio, el resultado será el que se obtenga de evaluar nuestra RC, `misremitentes`, la cual mira la dirección de correo-e del remitente (`check_sender_access` busca la dirección correo-e del remitente en la tabla) y su resultado es aceptar (si es de nuestro dominio o subdominios, en el patrón que hemos usado) o denegar; esta RC no puede devolver No-Sé, porque el último patrón empareja cualquier dirección.

Usamos una tabla tipo regexp (expresiones regulares) porque se examinan secuencialmente y permite poner al final una expresión *catch-all*. Con tablas indexadas creo que no habría manera de conseguir esto. Sin embargo hay una forma de hacerlo sólo con tablas indexadas:

main.cf

```
smtpd_restriction_classes = misremitentes
misremitentes = check_sender_access hash:/etc/postfix/misremitentes, reject
smtpd_sender_restrictions = check_client_access hash:/etc/postfix/misclientes
```

/etc/postfix/misremitentes

```
midominio.com OK
```

/etc/postfix/misclientes

```
midominio.com misremitentes
```

Pero en este caso el rechazo se haría en la restricción `reject`, que no devuelve la causa al cliente.

Mencionar por último que aunque Postfix decida en cualquier fase que no va a aceptar el mensaje, deja pasar por todas las fases y emite el error al final de la fase RCPT TO, lo cual nos puede despistar al hacer pruebas. Este comportamiento se puede anular con `smtpd_delay_reject = no`.

Lista de restricciones

Notacion	
D(IP)	Nombre de dominio por resolucion inversa de DNS de la direccion IP
[IP]	Esta IP o su red
[a,b]	Este dominio o su padre
[x]	El dominio x o dominios padre
	El dominio x o subdominios

Restricción	Arg	RC	Depende de	Explicación	Acción	Notas
Conexión de IP						
reject_unknown_client	IP	CHSR	-	IP no tiene PTR	Reject	
permit_mynetworks	IP	CHSR	$mynetworks	-	Permit	
check_client_access	IP	CHRS	map:client_access [IP],[D(IP)]			1,3
reject_maps_rbl	IP	CHSR	$maps_rbl_domains			2
HELO/EHLO h						
reject_invalid_hostname	h	HSR	-	Sintaxis incorrecta de 'h'	Reject	
permit_naked_ip_address	h	HSR	-	h es una direcciín IP sin []	Dunno	
reject_unknown_hostname	h	HSR	-	h no tiene en DNS registro A ni MX	Reject	
reject_non_fqdn_hostname	h	HSR	-	h no está en la forma FQDN	Reject	
check_helo_access	h	HSR	map:helo_access [h]			1
MAIL FROM a@b.c						
reject_unknown_sender_domain	b.c	SR	-	b.c no tiene registro A ni MX	Reject	
check_sender_access	a@b.c	SR	map:sender_access a@b.c,[b.c],a@			1,4
reject_non_fqdn_sender	b.c	SR	-	b.c no está en la forma FQDN	Reject	
reject_sender_login_mismatch						
RCPT TO d@e.f						
permit_mx_backup	e.f	R	-	Somos MX para e.f && Permit	Accept	
check_recipient_access	d@e.f	R	map:recipient_access d@e.f,[e.f],d@			1,4
reject_unknown_recipient_domain	e.f	R	-	e.f no tiene registro A ni MX	Reject	
reject_non_fqdn_recipient	e.f	R	-	e.f no está en forma FQDN	Reject	
permit_auth_destination	e.f	R	$relay_domains	e.f = <$relay_domains> && Permit	Dunno	
			$mydestination	e.f = $mydestination && Permit		
			$inet_interfaces	e.f = $inet_interfaces && Permit		
			$virtual_maps	e.f => $virtual_maps && Permit		
reject_unauth_destination	e.f	R	$relay_domains	e.f = <$relay_domains>	Reject	

			$inet_interfaces	e.f = $inet_interfaces && Dunno	
			$virtual_maps	e.f => $virtual_maps && Dunno	
check_relay_domains	IP, e.f	R	$relay_domains	D(IP) = <$relay_domains> && Permit	Reject
			$relay_domains	e.f = <$relay_domains> && Permit	
			$mydestination	e.f = $mydestination && Permit	
			$inet_interfaces	e.f = $inet_interfaces && Permit	
			$virtual_maps	e.f => $virtual_maps && Permit	
header_check, body_checks					

Notas

1 Si se encuentra en la tabla, aplicar el resultado que figure. Si no, No-Sé.

2 Si se encuentra, rechazar. Si no, No-Sé.

3 Si la dirección IP del cliente es `a.b.c.d` que se resulve al nombre de dominio `x.y.z`, Postfix busca en la tabla en el orden:

```
x.y.z
y.z
z
a.b.c.d
a.b.c
a.b
a
```

Se queda con la primera que encuentre. Si el tipo de la tabla es regexp o pcre, entonces se recorre secuencialmente y se intentan emparejar todas sus entradas con `x.y.x` y después con `a.b.c.d`

4 Si la dirección de correo-e `v@x.y.z` Postfix busca en la tabla en el orden:

```
v@x.y.z
x.y.z
y.z
z
v@
```

Se queda con la primera que encuentre. Si el tipo de la tabla es regexp o pcre, entonces se recorre secuencialmente y se intentan emparejar todas sus entradas con `v@x.y.z`

Si una restricción no devuelve aceptar o rechazar, el resultado es No-Sé.

Las restricciones con C pueden aparecer en la RC `smtpd_client_restrictions`
Las restricciones con H pueden aparecer en la RC `smtpd_helo_restrictions`
Las restricciones con S pueden aparecer en la RC `smtpd_sender_restrictions`
Las restricciones con R pueden aparecer en la RC `smtpd_recipient_restrictions`

Resumen

Tras cada fase del diálogo SMTP, las restricciones que figuren en

`smtpd_{client,helo,sender,recipient}_restrictions` son evaluadas secuencialmente. Tan pronto como una de ellas devuelve aceptar o rechazar, no se evalúan las siguientes y el resultado de la RC es ése. Si el resultado de una restricción es No-Sé, se evalúan las siguientes.

Si el resultado de una RC es aceptar o No-Sé, se aplicará la RC de la siguiente fase.

Si alguna restricción de una RC evalúa a rechazar, el mensaje será rechazado (tras la fase RCPT TO, excepto si se usa `$smtpd_delay_reject=no`).

Postfix obliga, por razones de seguridad, a incluir, en la RC `smtpd_recipient_restriction`, al menos una de las siguientes restricciones: `check_relay_domains`, `reject_unauth_destination` o `reject`, porque todas ellas devuelven rechazo si no se cumplen ciertas condiciones.

Luis Meléndez Aganzo - luism@uco.es
Septiembre 2002