Postfix, Courier y MySQL

Alvaro Marín Illera split77@terra.es

5 de Abril del 2002
Revisión: 21 de Enero del 2003

Instalación y configuración de Postfix, Courier y MySQL para operar juntos y conseguir un completo servidor de correo (SMTP e IMAP) con usuarios "virtuales" en una Base de Datos.

Introducción

Desde hace tiempo uso Postfix para el envío de correos con el fin de depender lo menos posible de otros servidores, que daban más de un problema (autentificación, caídas...).
Postfix es un MTA (Message Transfer Agent) relativamente fácil de administrar, seguro (al menos más que sendmail y otros) y que no sobrecarga mucho la máquina ya que solamente se cargan los módulos necesarios en cada momento. Su función es comunicarse con los otros servidores (rigel.deusto.es, terra.es ...), para entregarse entre ellos el correo.

Por ejemplo, cuando algún amigo nuestro nos quiere enviar un mail, se conecta al servidor de correo de terra (mailhost.terra.es) y mediante el protocolo SMTP, manda los comandos necesarios para que coja su mensaje. Posteriormente, mailhost.terra.es se conecta con rigel.deusto.es y le entrega dicho mail, con lo que éste lo guarda en el buzón del usuario al que iba dirigido el mensaje. Si bajamos al aula de correo de la Universidad, mediante el protocolo pop3 nos comunicamos con rigel.deusto.es para que nos devuelva los mensajes que hay en nuestro buzón y así recibir el mensaje que nuestro amigo nos había enviado.

Otro protocolo para poder "ver" los mensajes que tenemos en nuestro servidor, es IMAP (Internet Message Access Protocol), el cual es mucho más potente que el pop3 anteriormente citado. Permite entre otras muchas cosas, acceder a los mensajes del servidor como si fueran locales, creación de carpetas en el buzón del servidor, buzones accesibles por varias personas... He elegido courier-imap para implementarlo y poder ver así, toda su potencia :)

Por último, pero no por ello menos importante, ante cuestiones del estilo de "cómo se podría hacer para tener usuarios solo para el correo?", "cómo podría gestionar los usuarios del correo sin tener un /etc/passwd de 5000 líneas?"... decidí introducir también en la pareja postfix-courier, MySQL para tener una Base de Datos en la que se podrían ir guardando los usuarios "virtuales" (ya que no van a existir en el sistema y por tanto en /etc/passwd) de correo electrónico.

Como resumen a esta pequeña introducción, tenemos que Postfix se encargará de las labores de MTA, Courier de dar servicio IMAP y MySQL de almacenar los usuarios "virtuales".

Todo esto ha sido implementado sobre GNU/Linux Debian Woody en un Pentium75 con 32MB de RAM. Todos los comentarios de instalación y configuración los haré en base a esto, así que u os instalais Debian (qué mejor disculpa que esta, no? ;P) o le dais a google y rpm como loc@s. :)

Configurando Postfix

Postfix es muy sencillo de ponerlo a andar. Con la configuración que viene por defecto (si no recuerdo mal), ya es capaz de enviar mensajes. Para instalarlo en Woody, basta con:

```
apt-get install postfix
```

También vamos a instalar el soporte que tiene postfix para mysql:

```
apt-get install postfix-mysql
```

Para l@s que no usen debian, este soporte hay que darselo recompilando postfix con las opciones necesarias de la forma que muestro a continuación:

```
make -f Makefile.init makefiles \
'CCARGS=-DHAS_MYSQL -I/usr/local/mysql/include' \
'AUXLIBS=/usr/local/mysql/lib -lmysqlclient -lz -lm'
make && make install
```

Sustituyendo /usr/local por la ruta donde tengais dichos archivos en vuestra máquina. A continuación copiamos el archivo virtual del src de postfix al directorio de éste:

```
cp src/virtual/virtual /usr/libexec/postfix
```

Una vez instalado, vamos a /etc/postfix y con vuestro editor favorito, abrimos el archivo principal de configuración llamado main.cf, el cual está estupendamente comentado. Os pego a continuación el mío para que podais ir comparando y comento solamente los cambios importantes:

```
queue_directory = /var/spool/postfix
command_directory = /usr/sbin
daemon_directory = /usr/lib/postfix
mail_owner = postfix
myhostname = TARTALO.spliTer.net
mydomain = spliTer.net
myorigin = $myhostname
mydestination = tartalo.spliter.net, spliter.net
```

La siguiente opción nos permite elegir en qué tipo de "formato" postfix va a guardar los mensajes en el buzón de cada usuario. Si elegimos Mailbox, los mensajes se guardan en un único archivo en el que se van encolando los mensajes según van llegando al buzón. El formato Maildir, consiste en un directorio en el que hay tres subdirectorios en el que se van guardando los mensajes en diferentes archivos. Courier-IMAP necesita de esto, por lo que este es el que elegimos (nótese la "/" final):

```
home_mailbox = Maildir/
mail_spool_directory = /var/spool/mail/
relay_domains = $mydestination
mynetworks = 192.168.1.0/24
smtpd_banner = $myhostname ESMTP $mail_name
```

```
local_destination_concurrency_limit = 2
default_destination_concurrency_limit = 10
debug_peer_level = 2
debugger_command = PATH=/usr/bin:/usr/X11R6/bin xxgdb
$daemon_directory/$process_name $process_id & sleep 5
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
```

Es a partir de aquí donde cambia un poco la cosa. Vamos a ir comentando línea por línea:

Este parámetro, es lo que se le va a poner delante de un valor que tendremos en la BD para conseguir llegar hasta el lugar donde se guarda el buzón del usuario. Vamos a dejarlo con "/"

```
virtual_mailbox_base=/
```

Señalamos a postfix que los UserIDs y GroupIDs de los usuarios de correo los obtendrá por medio del archivo indicado, que accederá a MySQL

```
virtual_uid_maps=mysql:/etc/postfix/ids.cf
virtual_gid_maps=mysql:/etc/postfix/gids.cf
```

A continuación indicamos que mediante el archivo mysql_virt.cf vamos a acceder a MySQL para ver dónde están los buzones de los usuarios:

```
virtual_mailbox_maps=mysql:/etc/postfix/mysql_virt.cf
```

```
local_transport = virtual
program_directory = /usr/lib/postfix
```

Pasamos a continuación a mostrar el contenido de los ficheros antes referenciados en el main.cf llamados mysql_virt.cf, ids.cf y gids.cf. Todos ellos hacen referencia a una Base de Datos implementada en MySQL, que más adelante detallaremos. En este ejemplo, tanto postfix como MySQL está en la misma máquina, por ello el primero se conectará a la base de datos por medio de los sockets internos de unix/linux. Para ello ponemos el parametro hosts con el valor unix:mysqld.sock. Si estuviesen en máquinas distintas, habría que poner el nombre de la máquina en la que se encuentra MySQL.

mysql_virt.cf

```
#Usuario y password para acceder al servidor MySQL
user=postfix
password=postfix
#Nombre de la Base de Datos y la tabla
dbname=mail
table=passwd
#El campo que elegimos es maildir (lugar donde está el buzón), donde el id del usuario coincida con el destinatario del mensaje
select_field=maildir
where_field=id
#El host que tiene el MySQL
hosts=unix:mysqld.sock
```

ids.cf
#Usuario y password para acceder al servidor MySQL
user=postfix
password=postfix
#Nombre de la Base de Datos y la tabla
dbname=mail
table=passwd
#El campo que elegimos es uid
select_field=uid
where_field=id
#El host que tiene el MySQL
hosts=unix:mysqld.sock

gid.cf
#Usuario y password para acceder al servidor MySQL
user=postfix
password=postfix
#Nombre de la Base de Datos y la tabla
dbname=mail
table=passwd
#El campo que elegimos es gid
select_field=gid
where_field=id
#El host que tiene el MySQL
hosts=unix:mysqld.sock

También debemos modificar el archivo master.cf, añadiendo la línea:
virtual unix - n n - - virtual

Antes de que se nos olvide, debemos de cambiar también el archivo /etc/login.defs , descomentando la línea "QMAIL_DIR Mailidir" para indicar, que todos los usuarios que hagan login en nuestro sistema, usarán también el formato Maildir. Debemos de comentar las 2 que aparezcan a su lado, lógicamente.

Configurando MySQL

Hemos visto como Postfix va a hacer uso de MySQL. Vamos pues a instalarlo y configurarlo:

```
apt-get install mysql-server
```

No hay que tocar nada reseñable en la configuración del MySQL. Únicamente, si el servidor va a estar en otra máquina distinta a la máquina donde está postfix, tendremos que abrir un puerto para que pueda "escuchar". Por defecto, esto no es así debido a temas de seguridad. Para hacerlo, basta con comentar la línea skip-networking del fichero /etc/mysql/my.cfg y rearrancar el servidor.

Bien, una vez que lo tenemos instalado, vamos a crear la Base de Datos y la tabla necesaria para guardar toda la información de. Voy a mostrar cómo hacerlo paso por paso:

```
#Nos conectamos como root en un principio(la contraseña por defecto
está en blanco, por lo que bastaria con dar "enter")
mysql -h localhost -u root -p
#Creamos la BAse de Datos llamada mail
mysql>create database mail;
#Vemos si está creada:
mysql>show databases;
#Seleccionamos con la que vamos a trabajar:
mysql>use mail;
#Creamos la tabla en la que guardaremos la información de los usuarios
virtuales:
mysql>create table passwd(
id char(128)DEFAULT '' NOT NULL,
clear char(128)DEFAULT '' NOT NULL,
name char(128)DEFAULT '' NOT NULL,
uid int(10) unsigned NOT NULL,
gid int(10) unsigned NOT NULL,
home char(255)DEFAULT '' NOT NULL,
maildir char(255)DEFAULT '' NOT NULL,
KEY id (id(128))
);
#Vemos cómo ha quedado la tabla recién creada:
mysql> describe passwd;
```

Field	Type	Null	Key	Default	Extra
id	char(128)		MUL		
clear	char(128)				
name	char(128)				
uid	int(10) unsigned				
gid	int(10) unsigned				
home	char(255)				
maildir	char(255)				

```
#Creamos un usuario postfix y le damos todos los permisos en esta
tabla:
mysql>GRANT ALL ON mail.passwd TO postfix@localhost IDENTIFIED BY
"postfix";
#Dependiendo de la configuración, quizás haya que crear algún usuario
del estilo postfix@HOSTNAME... para que postfix pueda acceder desde
otras máquinas por ejemplo.
#Creamos tres entradas de ejemplo:
mysql>insert into passwd (id,clear,name,uid,gid,home,maildir)
values("ska@tartalo.spliter.net","ska","skalari","1003","8","/","/var/spool/mail/ska/Maildir/");
mysql>insert into passwd (id,clear,name,uid,gid,home,maildir)
values("punk@tartalo.spliter.net","punk","punkarra","1002","8","/","/var/spool/mail/punk/Maildir/");
mysql>insert into passwd (id,clear,name,uid,gid,home,maildir)
values("reggae@tartalo.spliter.net","reggae","rastaman","1004","8","/","/var/spool/mail/reggae/Maildir/");
```

```
#Vamos a hacer una select a ver qué es lo que tenemos
mysql> select * from passwd;
```

id	clear	name	uid	gid	home	maildir
ska@tartalo.spliter.net	ska	skalari	1002	8	/	/var/spool/mail/ska/Maildir/
punk@tartalo.spliter.net	punk	punkarra	1003	8	/	/var/spool/mail/punk/Maildir/
reggae@tartalo.spliter.net	reggae	rastaman	1004	8	/	/var/spool/mail/reggae/Maildir/

```
mysql>quit

### NOTA: para cambiar la password root, basta con hacer lo siguiente:
mysql> SET PASSWORD FOR root@localhost=password('nuevapass');
mysql> FLUSH PRIVILEGES;
> ### <code>
```

Bien, vamos a explicar todo esto un poco más despacio. Vemos que en la tabla hemos creado una serie de campos, entre ellos id que será el mail del usuario, clear que es su contraseña para autentificarse (puede cifrarse añadiendola de la forma password("contraseña")), uid que es el UserID y al que podemos ir asignándole números secuencialmente (para ello mirar antes /etc/passwd para ver a partir de cuál no está "ocupado"), el gid que en mi caso lo he puesto a 8 y que corresponde con el grupo de "mail" (lo he hecho así para que el usuario pueda escribir en el directorio /var/spool/mail que tiene como grupo propietario a "mail"), home que por ser usuarios "virtuales" lo dejamos a "/" mismo y el maildir o lugar donde se guardará el Maildir del usuario en cuestión. A esta ruta se le añade por delante el valor que habíamos puesto en /etc/postfix/main.cf como virtual_mailbox_base y que era "/". Se puede jugar combinando ambas, pero de esta forma aseguramos la ruta.

Vamos a hacer una pequeña modificación al script de arranque de MySQL, que se situa en /etc/init.d/mysql.
Modificaremos el apartado de "start)" para que quede de la siguiente forma:

```
'start')
# Start daemon
echo -n "Starting MySQL database server: mysqld"
/usr/bin/safe_mysqld > /dev/null 2>&1 &
for i in 1 2 3 4 5 6; do
if is_mysqld_alive; then break; fi
sleep 1
done
if is_mysqld_alive; then
ln -f /var/run/mysqld/mysqld.sock /var/spool/postfix/mysqld.sock
echo "."
else
```

```
echo "...failed."
fi
;;
```

Esto es debido a que el proceso smtpd de Postfix corre en una "jaula" o chroot y no puede acceder al socket, por ello creamos el enlace. Esta línea no haría falta en el caso de que accedamos a MySQL por red, pero en el caso que nos ocupa, hemos dicho que teniamos ambos servicios corriendo en la misma máquina.

Configurando Courier

Nos disponemos a configurar a continuación Courier. En sí, este programa es también un MTA, pero como hemos dicho, para esta función ya tenemos a Postfix. Vamos a usar solamente el courier-imap (también tiene courier-pop), para lo que hacemos :

```
apt-get install courier-imap courier-authdaemon courier-authmysql
courier-base
```

Con lo que se nos instalará la aplicación y el demonio de autentificación, la autentificación por MySQL... Los archivos de configuración, están situados en /etc/courier/ por lo que debemos dirigirnos allí.

Lo primero que debemos de cambiar es el modo de autentificación de los usuarios al usar IMAP. Para ello, editamos el archivo authdaemonrc y sustituimos la línea authmodulelist="authpam" por authmodulelist="authmysql", por lo que a partir de ahora, ya no se utilizará el módulo PAM, sino MySQL. Lógicamente hay que indicar a courier dónde debe mirar los usernames y passwords; para ello tenemos el archivo authmysqlrc, que queda de la siguiente forma:

```
MYSQL_SERVER TARTALO
MYSQL_USERNAME postfix
MYSQL_PASSWORD postfix
MYSQL_PORT 3306
MYSQL_OPT 0
MYSQL_DATABASE mail
MYSQL_USER_TABLE passwd
MYSQL_CLEAR_PWFIELD clear
DEFAULT_DOMAIN spliTer.net
MYSQL_UID_FIELD uid
MYSQL_GID_FIELD gid
MYSQL_LOGIN_FIELD id
MYSQL_HOME_FIELD home
MYSQL_NAME_FIELD name
```

MYSQL_MAILDIR_FIELD maildir

En el archivo imapd, tan solo hay que cambiar la última entrada y ponerla a YES (IMAPDSTART), que por defecto viene a NO.

Poniendo en marcha todo esto

Debemos asegurarnos que todos los demonios necesarios están corriendo en la máquina: courier-authdaemon, courier-imap, postfix y mysqld. Podemos ya entonces a hacer pruebas enviando mails a los usuarios creados en la Base de Datos. No debemos olvidar mirar los logs del sistema para ver los posibles fallos aparecidos. Cuando se envía un mail a un usuario, se deben de crear un directorio con su nombre en /var/spool/mail/ que contendrá el subdirectorio Maildir. Si esto no es así puede que sea por tema de permisos. Asegúrate de de que /var/spool/mail/ tenga como grupo propietario a "mail" (en mi caso) y tenga permisos de escritura.
Otro fallo que puede aparecer al enviar un mail es "Temporary lookup failure"; si nos fijamos en los logs vemos que postfix no puede conectarse a MySQL. Habrá que asegurarse entonces de que realmente nos podemos conectar con el usuario creado en postfix, que tiene los permisos necesarios...

Desde un cliente que soporte IMAP, podemos probar también que se leen bien los mensajes. Al autentificarse, el nombre de usuario, será el id de la tabla passwd de la Base de Datos y la contraseña será el campo clear.

Conclusiones

Con todo esto, en teoría ya deberías poder estar usando Postfix con MySQL y Courier, pero seguramente te encontrarás con más de un problema. Para poder solucionarlos, hay que estar continuamente mirando los logs para ver cuál puede ser la causa. En /var/log/mail.log aparece bastante información cada vez que se manda un mail y puede ayudar bastante.

A modo de resumen, podemos decir que el proceso es el siguiente: Cuando mandamos un mail por SMTP a un usuario, postfix lo recibe y lo primero que hace es ver si dicho usuario existe en la base de datos de MySQL. Por ello, se conecta a él y lanza una consulta SQL. Si no obtiene resultado, es que dicho user no existe por lo que no es posible enviar el mail. Si la consulta tiene éxito, coge el campo maildir cuyo id corresponde con el destinatario del mail, y mira a ver si existe el directorio Maildir en /var/spool/mail/usuario/. Si no es así lo crea y deposita el mail en dicho directorio, estando ya a disposición del

usuario para que lo consulte por vía IMAP a través de Courier.

Todo el tema de usuarios puede ser gestionado fácilmente mediante shell script o por PHP accediendo directamente a la Base de Datos, pudiendo añadir, eliminar, modificar...Lógicamente, a la tabla de usuarios se la puede añadir más campos de información sobre el usuario.

Postfix, Courier y MySQL: ¿hasta dónde quiere llegar hoy? ;)

Links

Mysql www.mysql.com
Postfix www.postfix.org
Courier www.courier-mta.org
Otros:
http://www.sweeney.demon.co.uk/pfix_imap_virtual.html
http://kummefryser.dk/HOWTO/mail/postfix_mysql.html

PD: gracias a pep-e del canal #debian-es de irc.freenode.net, por enviarme el mail escrito por Diego Berrueta en la lista de "debian-user-spanish" que comentaba el tema del chroot :)

Este documento ha sido escrito por un miembro de e-GHOST, y su contenido es libre de ser reproducido en otros medios bajo las condiciones de la Licencia FDL (GNU Free Documentation License).