

Un sistema de gestión de correo con software de fuentes abiertas

Carlos Terrón carlos.terron@gmail.com

16 de Noviembre 2002

Índice

1. Objetivos	4
2. La solución	5
2.1. El sistema operativo	5
2.2. El MTA	5
2.3. Servidor de POP3 e IMAP	6
2.4. Servidor de base de datos	6
2.5. Servidor de Web	7
2.6. Software auxiliar	8
2.6.1. Cache de DNS	8
3. Configuración del sistema operativo	8
3.1. Algunas notas de la instalación	9
3.2. Software que debe de instalarse	10
3.3. Asegurando el sistema	10
3.3.1. Parar servicios innecesarios	10
3.3.2. Revisar los usuarios	11
3.3.3. Configurar el acceso	12
3.4. Configuración de directorios y PATH	12
3.5. Quitar el MTA por defecto	13
4. Instalación de programas auxiliares	13
4.1. Daemontools	13
4.1.1. Una nota sobre los ficheros de logs	14
4.2. ucspi-tcp	14
5. Caché de DNS	14
5.1. Instalación	14
6. QMAIL	16
6.1. Instalación	16
6.2. Configuración	18
6.2.1. Configuración del MTA	18
6.2.2. Scripts de arranque	19
6.2.3. Instalación de los alias básicos	21
6.2.4. Protección antirelay	21
6.2.5. Arranque del sistema	22
6.3. Funcionamiento	23

7. Base de datos	24
7.1. Instalación	24
8. Vpopmail	25
8.1. Instalación	25
9. Courier IMAP	26
9.1. Instalación	26
10.Utilidades de mail	30
10.1. ezmlm	30
10.1.1. Instalación	30
10.2. Autoresponder	31
10.2.1. Instalación	31
11.Apache	31
11.1. Qmailadmin	33
12.SquirrelMail	34
12.1. Instalacion	34
12.2. Configuración	35
13.Administración del sistema	37
13.1. Gestión de dominios	37
13.2. Gestión de usuarios	38
13.3. Herramientas gráficas de administración	39
14.Introducción al correo de Internet	44
14.1. Direcciones	45
14.2. Protocolo SMTP	46
15.Qmail a fondo	47
15.1. Entrada de correo via SMTP	47
15.2. Entrada de correo via inyección local	48
15.3. Encolado de correo	51
15.4. Gestión de la cola de salida	51
15.5. Entrega de correo local	53
15.5.1. Extensiones de direcciones	53
15.5.2. Alias locales	55
15.6. Entrega de correo remoto	55
15.7. Asignación de direcciones de correo a usuarios	56
15.8. qmail y vpopmail	57

1. Objetivos

Se trata de implementar un sistema de correo completo usando software de fuentes abiertas. Se busca:

- Gestionar correo para varios dominios.
- Servir POP3 a varios dominios.
- Servir correo IMAP a varios dominios.
- Webmail.
- Utilidad de administración sencilla, a ser posible por web.
- Sistema veloz, seguro y escalable.

El corazón de todo el sistema es el software que se ocupa de la gestión de correo. Es conocido como MTA (*Mail Transport agent*). Es el encargado de recibir el correo, repartirlo en los buzones para aquellos dominios que gestiona, enviar el correo que recibe de aquellos clientes SMTP autorizados.

El correo que se almacena localmente debe de estar disponible a través de IMAP (*Internet Message Access Protocol*) y de POP3 (*Post office protocol version 3*). Es crucial que el almacenamiento local de correo sea *seguro* y que no se produzca problemas por acceso múltiples entre el MTA y los programas que gestionan el IMAP y POP.

Se quiere que el correo esté accesible a través de un webmail, se necesitará instalar un servidor de web. Para garantizar la seguridad de las comunicaciones este webmail debe de usar SSL (*Secure Socket Layer*).

Tradicionalmente en los sistemas Unix, los datos de los usuarios¹ están en el fichero `/etc/passwd`. Sin embargo, esto no es práctico para el sistemas de correo con muchos usuarios y varios dominios: no puede tenerse dos usuarios con el mismo nombre y se obliga a tener una cuenta local para cada uno. Aparte, la búsqueda en ficheros planos de texto es muy lenta. La solución a estos problemas es usar una base de datos SQL. Otra posible solución que puede estudiarse es usar un sistema de directorio tipo LDAP.

Actualmente la mayoría de los sistemas Unix tienen un mecanismo, denominado PAM (pluggable authentication module) que permite el uso de otros sistemas de autenticación distinto al fichero de `passwd`, pero manteniendo las tradicionales interfaces de Unix de programación.

¹nombre de la cuenta, clave, descripción

2. La solución

2.1. El sistema operativo

El sistema operativo tipo Unix es necesario para implementar esta solución. Tenemos varias alternativas, tanto de fuentes abiertas (Linux, FreeBSD, NetBSD, OpenBSD) como cerradas (Solaris, Aix). El sistema elegido es Linux: es de fuentes abiertas, tiene un buen rendimiento y buen soporte de hardware. Como distribución de Linux, se usará una *Fedora Core 1 y Debian Woody*. Con ligeras modificaciones, lo que se comenta aquí es aplicable a otras distribuciones Linux (Mandrake, Conectiva, Gentoo).

Debe de comprobarse que el hardware donde se va a instalar el equipo es compatible con la distribución de sistema elegida. En este documento se supone una instalación básica del sistema junto con las herramientas de desarrollo necesarias para poder compilar. También se supone que están actualizados los paquetes de la distribución adecuadamente con todas las erratas publicadas corregidas.

Hacer notar que se van a usar los paquetes estandar de las distintas distribuciones excepto el sistema de correo, el cual se compilará e instalará manualmente.

2.2. El MTA

Dentro del mundo Unix los principales MTA que se usan son *sendmail*, *postfix*, y *qmail*.

- *sendmail* Sendmail, es un clásico MTA, que viene de todos los Unix, tanto comerciales como libres. Es bastante rápido sacando correo. En contra tienen un mal historial de seguridad, un fichero de configuración críptico y una arquitectura monolítica. Puede encontrarse en <http://www.sendmail.org>.
- *postfix* Postfix que nace como sustituto del sendmail. Presenta una arquitectura modular (varios programas que se comunican entre ellos), intenta ser seguro, rápido y eficiente. De puertas afuera, intenta parecerse a sendmail en todo lo posible. Puede encontrarse en <http://www.postfix.org>.
- *qmail* Al igual que postfix, parte de una arquitectura modular, con un conjunto de programas que se comunican entre sí. Busca ser rápido, eficiente y seguro. Presenta una particular manera de almacenar el

correo, los Maildir, que permiten huir de los mbox que presentan problemas de corrupción. El único pero que puede ponerse es que no tiene un desarrollo continuo como postfix o sendmail. Puede encontrarse en <http://cr.yp.to/qmail.html>.

A sendmail le falta modularidad, su configuración es complicada y el uso de un formato de buzón no es interesante para este proyecto. Respecto a postfix, integra muchas de las cosas que en qmail hay que hacer a través de parches. Sin embargo la simplicidad de qmail y su velocidad, así como una base de código más pequeña le hace más sencillo su uso.

La elección de qmail como MTA condiciona el resto del software que se va a usar en esta implementación, puesto que para aumentar su eficiencia. Este software es:

- *Daemontools*: conjunto de programas, que sustituye a los tradicionales arranque de Unix desde el proceso init o bien desde los directorios en /etc/rc. Puede encontrarse en <http://cr.yp.to/daemontools.html>.
- *ucspi-tcp*: conjunto de programas, que implementan funcionalidades parecidas a inetd: escuchar en determinados puertos y lanzar procesos conectados con ellos. Puede encontrarse en <http://cr.yp.to/ucspi-tcp.html>.

Para apoyar el uso de dominios virtuales y no tener usuarios locales vamos a usar un software (que también puede usarse con postfix) llamado vpopmail. Este puede encontrarse en <http://inter7.com/vpopmail>. Este software también nos permite interactuar con LDAP en caso de que queramos tener a los usuarios en un servicio de directorio. Este programa va a ser el que se ejecute y acceda a base de datos para comprobar los usuarios.

2.3. Servidor de POP3 e IMAP

Como servidor de POP3 e IMAP se usará Courier - a pesar de que qmail trae un servidor de pop3 propio -. Se aprovechará los módulos de autenticación de Courier que permiten usar vpopmail - el software que se usa para gestionar dominios virtuales - para realizar la autenticación. El Courier IMAP puede bajarse de <http://www.inter7.com/courierimap>.

2.4. Servidor de base de datos

A la hora de manejar los datos de los usuarios de manera eficiente, vamos a usar una base de datos SQL. Esta base de datos dará soporte a las diferentes partes del sistema.

Dentro del mundo de fuentes abiertas nos encontramos con dos soluciones de base de datos muy usadas

- *MySQL* (<http://www.mysql.org>). Es una popular base de datos, muy usada en sitios web. Su mejor característica es la velocidad. Sin embargo, se han sacrificado por esto muchas características que son importantes en una base de datos: no soporta transacciones, los bloqueos son a nivel de tabla, no tiene procedimientos almacenados, vistas, o sigue de cerca el estandar ANSI. Al menos en la versión 3.x que es la más utilizada.
- *Postgresql* (<http://www.postgresql.org>). Una base de datos que orientada a objetos descendiente de los proyectos INGRES y POSTGRES desarrollados en Berkeley. Tienes muchas de las características que le faltan a MySQL, pero sin embargo no es tan rápida como ésta. También su manejo es más complicado que MySQL.

Los programas que se van a usar para implementar el sistema de correo soportan ambas bases de datos. A costa de algo de velocidad, postgresql es una base de datos más robusta que MySQL. Sin embargo, lo único que se necesita en el proyecto es una base de datos capaz de almacenar tablas planas. Además, el soporte de *vpopmailsoftware con el que se gestionará los dominios virtuales* de MySQL está más maduro que postgresql, con lo cual se usará MySQL para dar soporte a vpopmail.

2.5. Servidor de Web

El servidor de web va a permitir implementar dos funciones:

- Una interfaz de administración, con lo cual cualquiera con los permisos adecuados puede gestionar todo nuestro sistema de correo. Esto lo vamos a conseguir con la ayuda de un CGI llamado qmailadmin (<http://www.inter7.com/qmailadmin/>)
- Implementar un webmail: un sistema que permite enviar y recibir correo a través de un navegador web.

Dentro del mundo de fuentes abiertas el servidor Web por excelencia es Apache (<http://www.apache.org>). Es un programa altamente flexible, potente y rápido. Permite servir contenido estático, ejecutar CGIs, páginas dinámicas en diversos lenguajes (gracias a la cantidad de módulos que trae).

Por último se necesita el módulo de apache *mod_ssl* el cual permite soporte de SSL en las conexiones. Con él se conseguirá que las comunicaciones con el webmail y la parte de administración sean encriptadas.

2.6. Software auxiliar

2.6.1. Cache de DNS

Es un servidor DNS con la particularidad de que, cuando no conoce como dar servicio a una petición, la realiza, y la almacena en memoria, teniéndola disponible durante un cierto tiempo. La idea es no tener que efectuar peticiones a internet continuamente y que enlentezca el proceso. Las entradas se van descartando de la memoria por edad (o en caso de que se llene la cantidad de memoria asignada a almacén).

Tenemos varias opciones, entre las que destacan:

- **BIND** <http://www.isc.org/products/BIND/>. Es una implementación de los protocolos DNS, que se inició en Berkeley. Actualmente en su versión 9, aunque tanto las versiones 4 y 8 siguen usandose mucho. Es el software más usado como servidor de DNS. Sin embargo se ha visto afectado por diferentes problemas de seguridad en sus diferentes versiones. El uso como caché (en especial en las versiones 4 y 8) presenta problemas debido a que no se puede poner límite al tamaño de las peticiones, y además descarta las nuevas entradas en la caché en vez de las viejas cuando empieza a tener problemas de memoria.
- **Djbdns** <http://cr.yp.to/djbdns.html>. Del mismo autor que qmail, es un conjunto de programas que implementan los protocolos DNS. Uno de ellos es *dnscache*, el cual permitirá fácilmente implementar la caché DNS.

Por seguridad y facilidad de configuración, en este proyecto se usará el djbdns. Además se integra perfectamente con las herramientas que vamos a usar con qmail.

3. Configuración del sistema operativo

La tarea de configurar el sistema debe de hacerse con todo cuidado. Debe de instalarse con aquello que necesitemos exclusivamente, asegurándonos que esté a último nivel de parches, en especial de aquellos que afectan a la seguridad.

Las distribuciones pueden obtenerse de <http://fedora.redhat.com/> para el caso de Fedora y de <http://www.debian.org> para el caso de Debian. Es conveniente usar el mirror más cercano para bajarsela.

En este documento no se va a explicar los pasos para instalar las distribuciones Linux elejidas. Puede consultarse los manuales de instalación de Fedora

en <http://fedora.redhat.com/docs/>². Para la Debian los manuales de instalación se encuentran en <http://www.debian.org/releases/stable/installmanual>.

3.1. Algunas notas de la instalación

En esta sección se va a tratar de varios aspectos a tener en cuenta en la instalación.

- **Hardware soportado.** Debe de comprobarse que el hardware que va a usarse está soportado por la distribución.
- **Tamaño de particiones.** Sin contar el espacio necesario para el sistema, en caso de que se decida tener varias particiones, debemos de prestar especial atención a donde vayan a ir situados los directorios */var* y */home*. Bajo el primero va a ir situado la cola de correo y toda la gestión la información de actividades del sistema (logs). En el segundo directorio estarán situados los buzones de correo. Si se busca redundancia lo ideal es que estas dos particiones estén situadas en RAID 5 (aunque para crear un RAID 5 se necesitan al menos 3 discos). También si se busca la máxima concurrencia es interesante disponer las particiones en diferentes discos. En especial hay que tener en cuenta el tamaño asignado a */home*, puesto que es donde se van a almacenar los buzones de correo.
- **Parches de seguridad.** Desde que se publicaron las distribuciones han surgido problemas de seguridad que se han ido corrigiendo a través de diversos parches. Deben de actualizarse los programas instalados que hayan tenido problemas. Las actualizaciones de Fedora pueden encontrarse en <http://fedora.redhat.com/download/updates.html>, mientras que para Debian pueden buscarse en <http://www.debian.org/security/>. Ambas distribuciones tienen herramientas que permiten la actualización automática.
- **Usuarios.** Al ser una máquina destinada a servidor, los usuarios que tengan acceso a ellas deben ser los mínimos. Deben de elegirse las claves **robustas y complejas**. No conviene trabajar como root³ nada más que lo necesario, con lo cual se debe de crear otro usuario para poder manejar la máquina (y en caso de necesitar root usar su).

²Actualmente hay muy poca documentación sobre la Fedora. Puede mirarse sin embargo la de la Redhat 9 en <http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/>

³Usuario de administración

- **Interfaces de red.** Es ideal contar con dos interfaces de red. Esto nos permite tener una interfaz conectada a Internet y otra de administración.

3.2. Software que debe de instalarse

Ha continuación se detalla una lista de paquetes que vienen con la distribución y que deben de ser instalados, bien de los CDROM originales o bien de las actualizaciones. Es **importante** comprobar estas últimas.

- **Apache.** Servidor web.
- **OpenSSL.** Librerías criptográficas.
- **Soporte SSL Apache.**
- **OpenSSH.** Secure Shell. Permite acceder remotamente a la máquina a través de un canal cifrado.
- **MySQL.** Base de datos.
- **Software de desarrollo.** Este software solo lo necesitamos para compilar las cosas.

3.3. Asegurando el sistema

Esta sección se describe los pasos necesarios para asegurar el sistema base y dejar sólo lo imprescindible para dar el servicio adecuado.

3.3.1. Parar servicios innecesarios

En los sistemas Linux, hay varias maneras de arrancar servicios⁴. En primer lugar desde los scripts del sistema de arranque, desde los *super daemons*, como *xinetd* o *inetd*, o desde scripts propios.

Las distribuciones elegidas usan el llamado sistema de arranque de *System V*. En este modelo, existe un proceso llamado *init*, el cual es el padre de todos los procesos, y cuya configuración está almacenada en */etc/inittab*. En dicho fichero se define los *runlevels*, que no son más que una configuración que permite la existencia de un conjunto determinado de procesos. Para ver los servicios que se arrancan en un nivel determinado:

- En Fedora:

⁴Y no siempre es igual en todas las distribuciones.

`/usr/sbin/ntsysv`

- En Debian:

`/usr/bin/rcconf`

Debe de repasarse la configuración de xinetd o de inetd - en función del super daemon que se use. Para el primero, los ficheros de configuración están en `/etc/xinetd.d/` mientras que para el segundo el fichero es `/etc/inetd.conf`

Es importate saber si hay servicios a la espera de conexiones de Internet. Para ello podemos usar el comando `/usr/sbin/lsof -i -n`

3.3.2. Revisar los usuarios

Debemos de revisar la existencia de usuarios que acceden al sistema. En principio, si la instalación es limpia, no existirán más que los usuarios creados por ésta. En caso de no ser así debemos de repasar el fichero `/etc/passwd` viendo los usuarios creados después de la instalación. Podemos usar los comandos `/usr/sbin/userdel` para quitar los usuarios que no necesitamos. Tened precaución con los usuarios que se borran, puesto que hay una serie de ellos que son del sistema y necesarios para su funcionamiento (normalmente estos usuarios son los que tienen en el segundo campo del fichero `/etc/shadow` con `!!`)⁵.

Hay que comprobar también que no existan cuentas sin contraseña y que la robustez de las contraseñas usadas es adecuada. En el primero caso hay que repasar que el segundo campo del fichero `/etc/shadow` no esté vacío. En el segundo hay que asegurarse que el sistema de password que está en uso es md5. Para ello el segundo campo de las cuentas debe de comenzar por `1`. Ojo, las cuyo campo de password es `!!` están correctas. Simplemente son cuentas que no pueden usarse para hacer login en el sistema.

Por último puede comprobarse la robustez de las contraseñas con ayuda de un programa con el *John* el cual puede localizarse en <http://www.openwall.com/john/>

Normalmente, se crea una cuenta para la administración del correo. Esta cuenta recibirá los correos rebotados, los dirigidos al usuario root (Qmail nunca entrega correo al root de la máquina), y aquellos alias especiales, que se necesiten (como por ejemplo postmaster).

Si queremos que la cuenta se llame *mailadmin* debe de crearse y ponerle una clave con las siguientes órdenes:

⁵Este fichero sólo puede verse estando como root. Mirar formato en apéndice XXX

```
useradd mailadmin
passwd mailadmin
```

3.3.3. Configurar el acceso

Si la configuración elegida es tener dos interfaces de red, una puede usarse para administración, y para el tráfico de correo desde dentro de una empresa, con la otra interfaz abierta a Internet. Para acceder remotamente a la máquina se va a usar OpenSSH. Normalmente a la máquina sólo debe de tener acceso aquellos usuarios que deban administrarla, y ello no suele ser a través de Internet. Lo mismo es aplicable para la interfaz web de administración, aunque su configuración se tratará en el apartado de Apache.

Para cortar acceso al daemon de OpenSSH, llamado sshd, lo más sencillo es cortar el puerto 22 TCP en el firewall del kernel. Por ejemplo, si sólo se quiere que se acceda desde la subred 192.168.0.0/24, pues se utilizaría una orden como:

```
/sbin/iptables -A INPUT -p tcp -dport 22 -s ! 192.168.0.0/24 -j
DROP /etc/rc.d/init.d/iptables save
```

La segunda orden guarda la regla añadida en los ficheros de configuración del sistema.

Para buscar más información sobre el filtrado en el firewall puede consultarse la dirección web <http://www.iptables.org/documentation/HOWTO/es/packet-filtering-HOWTO.html>.

3.4. Configuración de directorios y PATH

Los programas se van a instalar en varios directorios. Para evitar tener que recordar largas rutas, conviene tener a los mismos de ejecutables en la variable PATH del sistema.

Debe de crearse un directorio donde vamos a instalar alguno de los programas que necesitemos. Así, los pasos a seguir:

1. Crear el directorio **/package** con las órdenes:

```
mkdir -p /package
chmod 1755 /package
```

2. Crear los directorios donde almacenaremos los logs de los diferentes daemons. Para ello deben de crearse los siguientes directorios:

/var/log/qmail/qmail	La información de ejecución del qmail
/var/log/qmail/qmail-smtp	La información de ejecución de la parte de qmail encargada
/var/log/qmail/pop3	La información de daemon de POP3
/var/log/qmail/imap	La información del daemon de IMAP

- Introducir en el PATH (siempre que no estén ya) del sistema los directorios **/usr/local/bin** y **/var/qmail/bin**. Para ello:

a) Editar el fichero */etc/profile* y al final añadir las líneas:

```
PATH=$PATH:/var/qmail/bin:/usr/local/bin
export PATH
```

b) Editar el fichero */etc/csh.login* y al final la línea:

```
setenv PATH "$PATH:/var/qmail/bin:/usr/local/bin"
```

3.5. Quitar el MTA por defecto

4. Instalación de programas auxiliares

4.1. Daemontools

La información sobre las daemontools pueden encontrarse en la página <http://cr.yp.to/daemontools.html>. La última versión en el momento de la escritura de este documento es la 0.76 y puede bajarse desde <http://cr.yp.to/daemontools/daemontools-0.76.tar.gz>.

Para instalar seguimos los siguientes pasos:

- Copiar y descomprimir el fichero en el directorio */package* (se descomprime con la ayuda de tar y gunzip: `gunzip -c daemontools-0.76 | tar xvf -`).
- Irse al directorio */package/admin/daemontools-0.76* y ejecutar **package/install**.

Con estos dos pasos debemos tener las daemontools instaladas y ejecutándose (fácil de comprobar con ejecutando un comando como *ps -ef | grep svscan* cuya salida debe ser similar a:

```
[terron@baldurgate]:admin>ps -ef | grep svscan
root 1102 1 0 Dec13 ? 00:00:00 /bin/sh /command/svscanboot
root 1111 1102 0 Dec13 ? 00:00:00 svscan /service
```

Ahora debe de crearse el directorio donde van a colocarse los diferentes scripts de arranque:

```
mkdir -p /service
chmod 1755 /service
```

4.1.1. Una nota sobre los ficheros de logs

El sistema de correo montado usa el programa multilog de las daemontools para escribir los logs a disco junto con una fecha del momento en el que se produjeron. Por defecto, el tamaño de los ficheros está limitado a 99999 bytes y conserva los 10 últimos. Sin embargo esto puede que no sea suficiente para sitios con mucho tráfico. Para ello, en los scripts que hagan uso de multilog se usan dos variables para poder controlar el tamaño y el numero de ficheros que se guardaran. Estas variables se llamaran MAXLOGSIZE y NUMLOG.

4.2. ucspi-tcp

La información referente a este paquete puede encontrarse en la página web <http://cr.yp.to/ucspi-tcp.html>. La instalación de la misma puede hacerse siguiendo los siguientes pasos:

1. Comprobar la última versión del paquete y bajar las fuentes. La última versión en el momento de escritura de este documento es la 0.88. Puede bajarse de <http://cr.yp.to/ucspi-tcp/ucspi-tcp-0.88.tar.gz>.
2. Descomprimir y desempaquetar el archivo: `gunzip -c ucspi-tcp-0.88.tar.gz | tar xvf -`
3. Irse al directorio resultante (`cd ucspi-tcp-0.88`). Ejecutar *make*. Como root, ejecutar en dicho directorio *make setup check*

Tras estos pasos tendremos las distintas utilidades instaladas en */usr/local*.

5. Caché de DNS

Se va a usar un caché de DNS para acelerar las consultas a DNS. El software elegido es el djbdns. Debe de tenerse instalada las daemontools para poder ejecutar correctamente este software.

5.1. Instalación

La página principal del Djbdns es <http://cr.yp.to/djbdns.html>. El software puede bajarse de la siguiente dirección: <http://cr.yp.to/djbdns/djbdns-1.05.tar.gz>. Para instalar este software deben de seguirse los siguientes pasos:

1. Descomprimir el fichero:

```
gunzip -c djbdns-1.05.tar.gz | tar xvf -
```

2. Ejecutar la siguiente orden para compilar el programa:

```
make
```

3. Como superusuario ejecutar la siguiente orden para instalar el programa (por defecto se instala en */usr/local*.

```
make setup check
```

4. Crear el directorio */etc/dnscache*

```
mkdir -p /etc/dnscache
```

5. Añadir los usuarios *dnscache* y *dnslog*. Para ello ejecutar las siguientes órdenes:

```
useradd dnscache
useradd dnslog
```

6. Ejecutar la siguiente orden para generar los ficheros de configuración del caché de DNS:

```
dnscache-conf dnscache dnslog /etc/dnscache
```

7. Crear el directorio */var/log/dnslog* y cambiarle el propietario al usuario *dnslog*. Para ello deben de ejecutarse las siguientes ordenes (como root):

```
mkdir -p /var/log/dnslog
chown dnslog /var/log/dnslog
```

8. Cambiar el sitio donde almacena los logs el caché. Para ello debe de cambiarse el script */etc/dnscache/log/run* al siguiente:

```
#!/bin/sh
MAXLOGSIZE=1048576
NUMLOG=7
exec setuidgid dnslog ssize $MAXLOGSIZE nnum $NUMLOG multilog t /var/log/dns
```

9. Crear los enlaces simbólicos en el directorio */service*.

```
ln -s /etc/dnscache /service/dnscache
```

10. Cambiar el fichero */etc/resolv.conf* de tal manera que solo quede la linea:

```
nameserver 127.0.0.1
```

11. Comprobar el correcto funcionamiento del caché. Para ello resolver algunos nombres con `dnsip`:

```
dnsip www.slashdot.org
dnsip www.barrapunto.com
```

En caso de que exista algún problema pueden consultarse los logs que está generando el caché en */var/log/dnscache/current*⁶

Es importante para el correcto funcionamiento del caché que en el PATH de búsqueda del sistema esté incluido el directorio */usr/local*. En caso de que no sea así siempre se puede editar los ficheros */etc/dnscache/run* y */etc/dnscache/log/run* para usar caminos absolutos hacia los ejecutables.

Debe de mirarse la sección del texto que explican el funcionamiento de las daemontools para usarlas correctamente y comprender exactamente el proceso de arranque de los servicios que las usan.

6. QMAIL

A lo largo de esta sección va documentarse la instalación, funcionamiento y configuración de qmail. Dada la particular configuración de arranque que va a usarse, deben de tenerse instalado las daemontools y ucspi-tcp para el correcto funcionamiento de qmail. °

6.1. Instalación

La página web del programa puede encontrarse en <http://cr.yp.to/qmail.html>. La última versión disponible a la hora de escribir este documento es la 1.03. La URL para bajarse el software es:

<http://cr.yp.to/software/qmail-1.03.tar.gz>.

Son necesarios dos parches para qmail. El primero para que reconozca la dirección 0.0.0.0 como local, y el segundo para que pueda compilarse con versiones recientes de la glibc (debido a un cambio en la implementación de `errno`),

⁶En el apendice XXX se hace referencia a este formato de logs

<http://www.suspectclass.com/~sgifford/qmail/qmail-0.0.0.0.patch>
<http://www.qmail.org/moni.csi.hu/pub/glibc-2.3.1/qmail-1.03.errno.patch>

A continuación se detallan los pasos de instalación, suponiendo que la misma empieza desde cero. Durante toda la instalación se supone que el software se instala en la ruta por defecto, es decir */var/qmail*.

1. Descomprimir el fichero:

```
gunzip -c qmail-1.03.tar.gz|tar xvf -
```

2. Aplicar los parches. Cambiar al directorio donde se ha descomprimido el software y ejecutar con cada uno de los parches:

```
patch -p1 < ruta_al_parche
```

3. Crear el directorio donde se instalará qmail:

```
mkdir -p /var/qmail
```

4. Crear los usuarios necesarios para la ejecución de qmail. Estos usuarios no deben de existir en el sistema, si se parte de una instalación de cero. Para ello debe ejecutarse como *root* la siguiente secuencia de ordenes:

```
groupadd nofiles  
useradd -M -g nofiles -d /var/qmail/alias alias  
useradd -M -g nofiles -d /var/qmail qmaild  
useradd -M -g nofiles -d /var/qmail qmail1  
useradd -M -g nofiles -d /var/qmail qmailp  
groupadd qmail  
useradd -M -g qmail -d /var/qmail qmailq  
useradd -M -g qmail -d /var/qmail qmailr  
useradd -M -g qmail -d /var/qmail qmails
```

5. Compilar e instalar el software:

```
make  
make setup check # Esta orden debe de hacer como root.
```

6. Desactivar el mta por defecto que trae la distribución y crear un enlace simbólico al programa que se encarga de inyectar el correo. Lo más sencillo es desactivar exim (mta por defecto de Debian) o sendmail (mta por defecto de Fedora) en los scripts de arranque con ayuda de ntsysv o rcconf.

```
cd /usr/sbin
mv sendmail sendmail.old
ln -s /var/qmail/bin/sendmail /usr/sbin/sendmail
```

Una vez compilado, la instalación genera la estructura de directorios de la siguiente tabla y copia los ficheros adecuados a cada localización.

/var/qmail/bin	Binarios
/var/qmail/alias	Directorio de alias del sistema
/var/qmail/boot	Contiene varios scripts de arranque
/var/qmail/control	Los ficheros de configuración
/var/qmail/doc	Documentación
/var/qmail/man	Páginas man
/var/qmail/queue	Cola de correo
/var/qmail/users	Base de datos de usuarios

6.2. Configuración

En esta primera fase de configuración se necesita el nombre completo de la máquina en Internet (conocido por las siglas FQDN). A lo largo de este documento se usará el dominio ficticio *webmail.reinos.org*. En caso de no disponer de un dominio, se puede usar la dirección IP de la máquina⁷. En el documento se usará la dirección *192.168.1.1*.

6.2.1. Configuración del MTA

El primer fichero importante a mirar es */var/qmail/control/me*. Tiene una sola línea donde está el FQDN de la máquina o bien la dirección principal de internet entre corchetes es decir:

```
webmail.reinos.org
o
[192.168.1.1]
```

El correo dirigido a *usuario@localhost* debe ser considerado como local. También debe de poder recibirse correo con un RCPT TO: *usuario@localhost*. Para ello debe de añadirse la línea *localhost* a los ficheros */var/qmail/control/locals* y */var/qmail/control/rcpthosts* (en caso de que no existan deben de crearse dichos ficheros). Deben de tener permisos adecuados (**chmod 644 fichero**).

⁷En caso de tener una varias interfaces de red, debe de usarse aquella que esté abierta a Internet

6.2.2. Scripts de arranque

El arranque se integra con las daemontools. Concretamente una de las utilidades, denominada *supervise*, es la encargada de monitorizar los daemons de qmail, rearrancándolos en caso de caída o error. Para ello explora una serie de directorios con una estructura particular⁸.

- Crear los directorios de los servicios. Para ello se ejecutan la siguiente secuencia de órdenes como root:

```
mkdir -p /var/qmail/supervise
mkdir -p /var/qmail/supervise/qmail
mkdir -p /var/qmail/supervise/qmail/log
chmod 1755 /var/qmail/supervise/qmail
mkdir -p /var/qmail/supervise/qmail-smtpd
mkdir -p /var/qmail/supervise/qmail-smtpd/log
chmod 1755 /var/qmail/supervise/qmail-smtpd
```

- Crear los directorios donde se guardarán los archivos de registro. Para ello ejecutar las siguientes órdenes como root:

```
mkdir -p /var/log/qmail
mkdir -p /var/log/qmail/qmail
mkdir -p /var/log/qmail/qmail-smtpd
chown qmail /var/log/qmail/qmail
chown qmail /var/log/qmail/qmail-smtpd
```

- Crear los siguientes scripts con el contenido indicado:

- */var/qmail/rc*

```
#!/bin/sh
exec env - PATH="/var/qmail/bin:$PATH" \
qmail-start ./Maildir/ 2>&1
```
- */var/qmail/supervise/qmail/run*

```
#!/bin/sh
exec /var/qmail/rc
```
- */var/qmail/supervise/qmail/log/run*

⁸Puede encontrarse más información en <http://cr.yp.to/daemontools/faq/create.html#runlog>

```
#!/bin/sh
MAXLOGSIZE=1048576
NUMLOG=7
exec /usr/local/bin/setuidgid qmaill \
/usr/local/bin/multilog ssize $MAXLOGSIZE nnum \
$NUMLOG t /var/log/qmail/qmail
```

- */var/qmail/supervise/qmail-smtpd/run*

```
#!/bin/sh
QMAILDUID='id -u qmaild'
NOFILESGID='id -g qmaild'
MAXMEM=4000000
exec /usr/local/bin/softlimit -m $MAXMEM \
/usr/local/bin/tcpserver -l 0 -H -v -p \
-x /etc/tcp.smtp.cdb \
-u $QMAILDUID -g $NOFILESGID 0 smtp \
/var/qmail/bin/qmail-smtpd 2>&1
```

Este script tiene dos particularidades: establece un máximo de *bytes* de memoria que puede consumir el proceso *tcpserver*, y por tanto el proceso *qmail-smtpd*. En este caso está fijado a 4 MB. Puede ajustarse asignando el valor requerido a *MAXMEM* en el script. La otra particularidad es el fichero */etc/tcp.smtp.cdb* del cual se hablará un poco más adelante.

- */var/qmail/supervise/qmail-smtpd/log/run*. Este script es el encargado de recoger los logs generados por *qmail-smtpd*. Por defecto, el programa que escribe los logs a disco guarda 10 y los rota cuando llega a 99999, lo cual es insuficiente para un sitio con mucho tráfico.

```
#!/bin/sh
MAXLOGSIZE=1048576
NUMLOG=7
exec /usr/local/bin/setuidgid qmaill \
/usr/local/bin/multilog ssize\ $MAXLOGSIZE nnum $NUMLOG \
t /var/log/qmail/smtpd
```

Una vez creado los scripts debe de dársele a los mismos permisos de ejecución con la orden:

```
chmod 755 script
```

6.2.3. Instalación de los alias básicos

Para tener correctamente configurado el sistema debe de existir la dirección de correo *postmaster*. También hay que generar alias para las cuentas *root* y *MAILER – DAEMON*. En la configuración propuesta la cuenta de usuario *mailadmin*. Para ello deben de crearse los siguientes ficheros, todos deben de tener el mismo contenido (*ℰmailadmin@localhost*) y darle permisos adecuados con la orden **chmod 644 fichero**.

```
/var/qmail/alias/.qmail-mailer-daemon
/var/qmail/alias/.qmail-root
/var/qmail/alias/.qmail-postmaster
```

Como se va a usar entrega de correo local debe de crearse el Maildir (el tipo de buzón de correo que vamos a usar) en la cuenta *mailadmin*. Para ello ejecutamos la orden:

```
/var/qmail/bin/maildirmake ~mailadmin/Maildir
```

6.2.4. Protección antirelay

Hoy en día el correo basura (spam) se ha convertido para mucha gente un problema. Por defecto, qmail no realiza relay de correo, es decir, no recibe un correo de un dominio externo y lo manda a otro dominio externo. Sin embargo, en un momento dado, puede que se quiera aprovechar el servidor de correo smtp para la empresa. Para ello va a crearse el fichero, *tcp.smtp*, el cual se convertirá a un formato especial con ayuda del programa *tcprules* (del paquete *ucspi-tcp*), y este fichero generado, va a pasarse como parámetro en los scripts de arranque de *qmail-smtpd* (como puede verse en el fichero */var/qmail/supervise/qmail-smtpd/run*).

Por ejemplo, si se quiere que el sistema de correo acepte relay desde la subred 10.0.0.0/24 y desde la máquina local (localhost, identificado normalmente por la red 127.0.0.0/8), tendrá que crearse el siguiente fichero *tcp.smtp*:

```
127.:allow,RELAYCLIENT=""
10.0.0.:allow,RELAYCLIENT=""
```

Posteriormente se procesa dicho fichero para crear */etc/tcp.smtp.cdb*, que es el fichero que va a usar el programa *tcpserver* para establecer la variable *RELAYCLIENT*, que hace que *qmail-smtpd* ignore el fichero */var/qmail/control/rcpthost* y permita el relay a las direcciones autorizadas. Para crear dicho fichero se ejecuta la orden:

```
tcprules /etc/tcp.smtp.cdb ficherotemporal < /etc/tcp.smtp
```

Esta orden debe de repetirse cada vez que se cambia el fichero */etc/tcp.smtp* y rearrancarse el proceso que lo usa (se verá más adelante como se realiza esto).

El formato que usa el fichero */etc/tcp.smtp* puede encontrarse en <http://cr.yp.to/ucspi-tcp/tcprules.html> y se describe en el apéndice XXX.

6.2.5. Arranque del sistema

qmail se arranca a través de svscan. Para ello, es necesario poner los scripts creados en los apartados anteriores bajo el control del mismo:

```
ln -s /service/qmail /var/qmail/supervise/qmail
ln -s /service/qmail-smtpd /var/qmail/supervise/qmail-smtpd
```

Con estas instrucciones debe de estar arrancado qmail. Debe de probarse que funciona adecuadamente. Para ello debe de tenerse instalado un cliente de correo que entienda el formato Maildir, como por ejemplo mutt.

Para probar, debe de entrar en el sistema a través de la cuenta mailadmin. Una vez dentro, configuramos mutt para que busque el correo en el Maildir. Para ello basta con asignar a la variable *MAIL* la ruta completa hacia el Maildir. Por ejemplo, usando bash como shell:

```
export MAIL="$HOME/Maildir"
```

Para probar si el sistema entrega el correo local correctamente puede ejecutarse la orden:

```
echo to: mailadmin | /var/qmail/bin/qmail-inject
```

Con mutt puede comprobarse si ha llegado correctamente el mail y comprobar que en el fichero de log del qmail, */var/log/qmail/qmail/current*.

Para comprobar que funciona correctamente los errores en caso de no existencia de una cuenta local, puede mandarse un correo a una cuenta no existente con la orden:

```
echo to: noexiste | /var/qmail/bin/qmail-inject
```

Puede comprobarse con el mutt que hemos recibido un mensaje de respuesta de qmail indicando la condición de error. En el mismo fichero de error que el caso anterior.

Para comprobar que funciona correctamente el envío a Internet, se necesita una cuenta de correo remota. Si la cuenta es *yo@unlugarlejano.com*, puede probarse la siguiente orden.

```
echo to: yo@unlugarlejano.com | /var/qmail/bin/qmail-inject
```

Puede comprobarse en el buzón remoto se ha recibido el correo. Puede también revisarse los logs en el mismo sitio anterior.

Por último hay que comprobar que funciona correctamente la cuenta de *postmaster*. Para se manda un mail local al postmaster de la siguiente manera:

```
echo to: postmaster | /var/qmail/bin/qmail-inject
```

Como se ha configurado la cuenta *mailadmin*, ese mail debe de llegar a dicha cuenta.

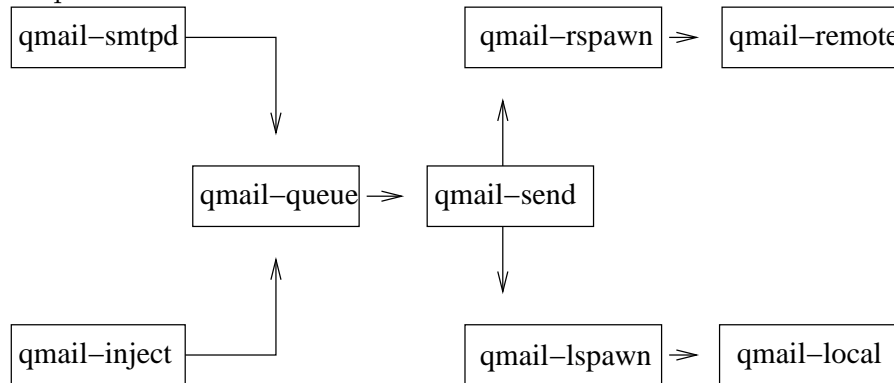
En caso de existir problemas, debe de comprobarse los ficheros de log de qmail.

6.3. Funcionamiento

En esta sección se hace un pequeño resumen de la estructura y funcionamiento de qmail. Los diferentes componentes de qmail son:

- **qmail-smtpd:** Se encarga de recibir el correo a través del protocolo SMTP. Normalmente es lanzado de un superserver como inetd, xinetd o tcpserver (el que se usará). Además se encarga del control de relay.
- **qmail-inject:** Se encarga de añadir y reescribir las cabeceras de los mensajes que le llegan a través de la entrada estándar y de pasar ese mensaje al qmail-queue. También se encarga de tratar las direcciones de correo.
- **qmail-queue:** Se encarga de meter el correo en la cola de salida para su posterior envío por qmail-send.
- **qmail-send:** Se encarga de procesar la cola de salida.
- **qmail-lspawn:** Se encarga programar las entregas de correo local.
- **qmail-local:** Se encarga de la entrega de correo local. Su comportamiento lo puede controlar el usuario.
- **qmail-rspawn:** Se encarga de programar las entregas de correo remoto.
- **qmail-remote:** Se encarga de enviar un mail a través de SMTP.

A continuación se incluye un gráfico de como se interrelacionan los diferentes procesos:



7. Base de datos

La base de datos usada va a ser MySQL. En ella se almacenará la información necesaria de los dominios y usuarios.

7.1. Instalación

Hay que instalar tanto el motor de base de datos como el soporte necesario para desarrollar. Esto debe hacerse antes de compilar el vpopmail.

Lo primero que debe de hacerse una vez instalado es darle una password de administrador al servidor. Una vez que el servidor de base de datos esté instalado, debemos de asegurarnos que el servidor está arrancado⁹ y darle una password al usuario de root:

```
mysqladmin password passwordnueva
```

Donde *passwordnueva* es la clave que se le da al administrador de la misma.

Ahora debe de crearse la base de datos que va a usarse para el vpopmail, crear el usuario para que el vpopmail acceda a la misma y darle los permisos adecuados:

1. Conectarse a la base de datos con el usuario de root. El sistema preguntará por la clave dicho usuario y una vez que se haya autenticado correctamente, nos presentará el prompt de mysql:

```
mysql -u root -p
```

⁹En Redhat se arranca con `/etc/init.d/mysqld start`

2. Se crea la base de datos y el usuario, con las siguientes ordenes en el prompt de mysql. Si el usuario que accede a la base de datos es *vpopmail* y la clave del mismo es *myclave*

```
create database vpopmail;  
grant all on vpopmail.* to vpopmail identified by 'myclave';  
flush privileges;
```

8. Vpopmail

Vpopmail es un software que permite la gestión de dominios virtuales. Este software va a usar MySQL como almacén de datos.

8.1. Instalación

Excepto la descompresión de las fuentes, todo el proceso de instalación de este componente de software debe de hacerse como root. El programa puede bajarse desde <http://vpopmail.sf.net>¹⁰

1. Bajar las fuentes de vpopmail.
2. Descomprimir las fuentes.

```
gunzip -c vpopmail-5.4.0.tar.gz | tar xvf -
```

3. Crear el usuario y grupo necesario para el funcionamiento del vpopmail:

```
groupadd vchkpw  
# Para el caso de Redhat  
useradd -M -s /sbin/nologin -g vchkpw vpopmail  
# Para el caso de Debian  
useradd -M -s /bin/false -g vchkpw vpopmail
```

4. Configurar, compilar e instalar el vpopmail con soporte para MySQL. Esta orden instalará por defecto en /home/vpopmail

```
./configure --enable-auth-module=mysql  
make  
make install
```

¹⁰La última versión estable es la 5.4.0

5. Configurar el acceso a la base de datos (login, password y base de datos) en el fichero `/home/vpopmail/etc/vpopmail.mysql`

```
localhost|0|vuser|vclave|vdatabase
```

Donde *vuser* es el usuario para acceder a la base de datos, *vclave* es la clave de dicho usuario y *vdatabase* es la base de datos que se va a usar.

Tras seguir este proceso, el vpopmail quedará instalado en `/home/vpopmail`. La estructura de directorios que queda es:

<code>/home/vpopmail/bin</code>	Programas ejecutables
<code>/home/vpopmail/doc</code>	Documentación
<code>/home/vpopmail/domains</code>	Almacen de correo
<code>/home/vpopmail/etc</code>	Ficheros de configuracion
<code>/home/vpopmail/include</code>	Includes para desarrollo
<code>/home/vpopmail/lib</code>	Librerias para desarrollo

9. Courier IMAP

Este programa va a ser el encargado del protocolo IMAP. Tiene varios módulos de autenticación. Uno de estos módulos permite usar las librerías de vpopmail para obtener la información necesaria. Además, tiene un módulo que permite cachear la información de autenticación ahorrando consultas innecesarias que acaban en la base de datos.

9.1. Instalación

Debe de bajarse el courier-imap de la url <http://prdownloads.sourceforge.net/courier/courier-imap-1.7.0.tar.bz2>. La configuración de este software no puede hacerse como root. Para ello debe usarse una cuenta normal. Puede usarse la cuenta *mailadmin* para ello. La versión a la hora de escribir este documento es la 2.2.1.

En principio este software se va a instalar por defecto en `/usr/lib/courier-imap`. Los pasos a seguir son lo siguientes:

- Con una cuenta *distinta a root*, descomprimir el fichero, configurar y compilar el programa:

```
bzcat courier-imap-2.2.1.tar.bz2 | tar xvf -
cd courier-imap-2.2.1
./configure --with-authpam \
--sysconfdir=/etc/courier \
```

```
--with-authvchkpwd \
--without-authldap \
--without-authmysql
make
\end{quote}
```

En el caso de estar instalándose sobre Fedora, añadir la opción:
\begin{verbatim}--with-redhat

- Instalar como root, los binarios (desde el directorio donde se ha compilado el programa).

```
make install
```

- Crear el fichero que arrancará el authdaemon, el programa que se encargará de la autenticación. Para ello crear el fichero */etc/init.d/authdaemon*, con el siguiente contenido:

```
#!/bin/sh
#
# chkconfig: 345 95 95
# description: authdaemon
# processname: authdaemon
prefix=/usr/lib/courier-imap

if [ ! -f /etc/courier/authdaemonrc ]; then
exit 1
fi
if [ ! -x $prefix/libexec/authlib/authdaemon.plain ]; then
exit 1
fi

. /etc/courier/authdaemonrc
case "$1" in
start)
$prefix/libexec/authlib/authdaemon.plain start
;;
stop)
$prefix/libexec/authlib/authdaemon.plain stop
;;
*)
echo "Usage: authdaemon start|stop"
```

```
exit 1
esac
```

Darle permiso de ejecución con:

```
chmod 755 /etc/init.d/autjdaemon
```

En el caso de la Fedora puede añadirse a los scripts de inicio con la orden:

```
/sbin/chkconfig --add authdaemon
```

- Crear el directorio donde van a almacenarse los logs de imap. Este directorio será */var/log/qmail/imapd*. Como root ejecutar las siguiente órden:

```
mkdir -p /var/log/qmail/imapd
```

- Crear el usuario para los logs, que llamaremos *imaplog*. Cambiar el propietario del directorio del paso anterior a este usuario. Como root ejecutar:

```
useradd -M -d /var/log/qmail/imapd imaplog
useradd -M -d /var/log/qmail/pop3d pop3log
useradd -M imapd
useradd -M pop3d
chown imaplog /var/log/qmail/imapd
chown pop3log /var/log/qmail/pop3d
```

- Crear los directorios para el servicio y para los logs. Establecer los permisos adecuados:

```
mkdir -p /usr/lib/courier-imap/supervise/imap
mkdir -p /usr/lib/courier-imap/supervise/imap/log
chmod 1755 /usr/lib/courier-imap/supervise/imap/log
mkdir -p /usr/lib/courier-imap/supervise/pop3
mkdir -p /usr/lib/courier-imap/supervise/pop3/log
chmod 1755 /usr/lib/courier-imap/supervise/pop3/log
```

- Crear el script */usr/lib/courier-imap/supervise/imap/run* con el siguiente contenido:

```
#!/bin/sh
USERIMAP='id -u imapd'
GROUPEMAP='id -g imapd'
exec \
/usr/local/bin/tcpserver -u "$USERIMAP" -g "$GROUPEMAP" \
-h -R 0 imap \
/usr/lib/courier-imap/sbin/imaplogin \
/usr/lib/courier-imap/libexec/authlib/authdaemon \
/usr/lib/courier-imap/bin/imapd Maildir 2>&1
```

- Crear el script */usr/lib/courier-imap/supervise/imap/log/run* con el siguiente contenido:

```
#!/bin/sh
exec \
/usr/local/bin/setuidgid imaplog \
/usr/local/bin/mtailog t \
/var/log/qmail/imapd
```

- Crear el fichero */usr/lib/courier-imap/supervise/pop3/run*

```
#!/bin/sh
USERIMAP='id -u pop3d'
GROUPEMAP='id -g pop3d'
exec \
/usr/local/bin/tcpserver -u "$USERIMAP" -g "$GROUPEMAP" \
-h -R 0 pop3 \
/usr/lib/courier-imap/sbin/pop3login \
/usr/lib/courier-imap/libexec/authlib/authdaemon \
/usr/lib/courier-imap/bin/pop3d Maildir 2>&1
```

- Crear el fichero */usr/lib/courier-imap/supervise/pop3/log/run*:

```
#!/bin/sh
exec \
/usr/local/bin/setuidgid pop3log \
/usr/local/bin/mtailog t \
/var/log/qmail/pop3d
```

- Dar permisos de ejecución a los scripts anteriores y añadir al sistema de arranque del supervise:

```

chmod 755 /usr/lib/courier-imap/supervise/imap/run
chmod 755 /usr/lib/courier-imap/supervise/imap/log/run
ln -s /usr/lib/courier-imap/supervise/imap /service/imap
chmod 755 /usr/lib/courier-imap/supervise/pop3/run
chmod 755 /usr/lib/courier-imap/supervise/pop3/log/run
ln -s /usr/lib/courier-imap/supervise/pop3 /service/pop3

```

- Quitar los permisos en los directorios que se han dado para compilar.

```

chmod 700 /home/vpopmail/lib
chmod 600 /home/vpopmail/lib/libvpopmail.a

```

En caso de problemas debe de mirarse si el proceso ha arrancado correctamente y los archivos de registro creado por el mismo.

10. Utilidades de mail

En esta sección se hablará de como instalar algunas de las utilidades que nos aportarán funcionalidad a todo el sistema: gestores de listas de correos y autoresponders.

10.1. ezmlm

Este es un software del mismo autor del qmail, que permite una fácil creación de listas de correo. Permite que cada usuario sea capaz de crear una lista sin tener que estar molestando a los administradores del sistema.

10.1.1. Instalación

La última versión disponible es la 0.53, que puede bajarse de la url <http://cr.yp.to/software/ezmlm-0.53.tar.gz>. Los ficheros de configuración por defecto son *conf-qmail*, *conf-bin* y *conf-man*, que indican donde está situado qmail, donde irán los binarios del gestor de lista y donde irán las páginas de manual. En este trabajo se quedará todo en su sitio por defecto.

Para instalar el programa deben seguirse los siguientes pasos:

- Descomprimir las fuentes, pasar al directorio resultante y compilar las fuentes:

```

gunzip -c ezmlm-0.53.tar.gz | tar xvf-
cd ezmlm-0.53
make
make man

```

- Como *root*, instalar los binarios:

```
make setup
```

10.2. Autoresponder

Este pequeño programa sirve para generar respuestas automáticas junto con Qmail.

10.2.1. Instalación

El programa puede bajarse de la url <http://www.inter7.com/devel/autorespond-2.0.2.tar.gz>. Necesita saber la localización del directorio de Qmail. Sin embargo, como se está usando la instalación por defecto, no es necesario modificar nada. El programa se instalará por defecto en */usr/local/bin*. Para instalarlo deben de seguirse los siguientes pasos:

- Descomprimir las fuentes, cambiar al directorio resultante y compilar las fuentes:

```
gunzip -c autorespond-2.0.2.tar.gz | tar xvf -  
make
```

- Como *root*, instalar el programa

```
make install
```

11. Apache

Apache es el servidor web que se va usar en el proyecto para dar soporte a un webmail y a una pequeña utilidad de administración de los diferentes dominios. Hay que asegurarse que se está instalando la última versión disponible para la distribución, puesto que hay ciertos problemas de seguridad serios que están resueltos en ellos.

En esta sección también se describe como instalar el cgi *qmailadmin* que va a usarse para configurar los dominios virtuales una vez creados.

Debe de decidirse también el puerto donde va escuchar el servicio de administración. Es aconsejable, si se dispone de dos interfaces de red, una abierta al exterior y otra de administración, la parte que se encarga de la administración debe de escuchar en la interna.

En esta sección no entrará en profundidad en las diferentes opciones de configuración de Apache, puesto que la complejidad del mismo puede abarcar más de un libro. Toda la documentación del servidor web puede encontrarse en la <http://httpd.apache.org/docs/>.

Se van a crear dos sitios web, uno de ellos se utilizará para el webmail, mientras que el otro se utilizará para la administración del sitio. En este ejemplo en la web principal va a estar el webmail y la zona de administración se va configurar en otra dirección IP y en un puerto distinto del 80. Los datos pueden consultarse en la maqueta.

El fichero de configuración de Apache se encuentra en la Redhat 7.3 en */etc/httpd/conf/httpd.conf*. Este archivo viene por defecto viene configurado con un montón de opciones para la Redhat. Se comentarán las opciones que deben añadirse o modificarse.

Los pasos a seguir para configurar Apache son los siguientes.

- Crear los directorios donde va a colocarse el webmail, la utilidad de administración y los ficheros de logs. Los pasos a dar son:

```
mkdir -p /var/log/httpd
mkdir -p /var/log/httpd/webmail
mkdir -p /var/log/httpd/qmailadmin
mkdir -p /var/www/webmail
mkdir -p /var/www/qmailadmin
mkdir -p /var/www/qmailadmin/cgi-bin
```

- Modificar la parte global de Apache con las opciones siguientes:
 - Añadir las siguientes líneas para que escuche en los puertos necesarios:

```
Listen 80
Listen 192.168.0.1:2000
```

- Cambiar la raíz del documento al webmail. Para ello, localizar la línea:

```
DocumentRoot "/var/www/html"
    cambiarla a
DocumentRoot "/var/www/webmail"
    Localizar las línea
<Directory "/var/www/html"> y cambiarla por
<Directory "/var/www/webmail">.
```


Con esto se consigue que el site principal de Apache sea donde va a estar situado el webmail.

- Añadir un servidor virtual nuevo en el puerto 2000 y escuchando en la IP interna. Para ello, al final del fichero añadir lo siguiente:

```
<VirtualHost 192.168.0.1:2000>
SSLProtocol all
    DocumentRoot "/var/www/qmailadmin"
    ErrorLog "/var/log/httpd/qmailadmin/error_log"
    TransferLog "/var/log/httpd/qmailadmin/access_log"
    ScriptAlias /cgi-bin/ /var/www/qmailadmin/cgi-bin/
    <Directory "/var/www/qmailadmin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>
    ServerName 192.168.0.1
</VirtualHost>
```

11.1. Qmailadmin

Este CGI puede obtenerse en <http://www.inter7.com/qmailadmin.html>. Esta utilidad permite gestionar los diferentes dominios que se han creado. La url para la versión disponible en el momento de escribir este documento es <http://www.inter7.com/qmailadmin/qmailadmin-1.0.6.tar.gz>

Para instalarlo deben de seguir los siguientes pasos:

1. Descomprimir las fuentes. Cambiar al directorio donde se han descomprimido:

```
gunzip -c qmailadmin-1.0.6.tar.gz | tar xvf -
cd qmailadmin-1.0.6
```

2. Configurar y compilar el programa, teniendo en cuenta las opciones que se han creado para Apache en el sitio de administración, es decir que el raíz del servicio web virtual está en /var/www/qmailadmin y que los cgi para dicho servicio están en /var/www/qmailadmin/cgi-bin:

```
./configure --enable-htmdir=/var/www/qmailadmin \
            --enable-cgibindir=/var/www/qmailadmin/cgi-bin
make
```

3. Como root, instalar el programa:

```
make install
```

El programa de administración podrá accederse a través de `http://192.168.0.1/cgi-bin/qmailadmin`. Para entrar basta con introducir el dominio y la clave que se la asignado al postmaster.

12. SquirrelMail

La aplicación de webmail que va a usarse está escrita completa en PHP. Para consultar el correo que se mostrará a los usuarios utiliza IMAP, mientras que para enviar, usa SMTP a través del propio Qmail instalado. Este modo de funcionamiento tiene la ventaja de que puede separarse fácilmente la parte de webmail a otra máquina.

12.1. Instalacion

Debe de bajarse la última versión disponible, que puede encontrarse en <http://www.squirrelmail.org/download.php>. En el momento de escribir este documento la versión disponible es 1.2.11.

El punto donde va a instalarse es `/var/www/webmail`, estando ya Apache configurado para trabajar desde ese directorio.

Los pasos a seguir por la instalación son:

- Descomprimir las fuentes:

```
gunzip -c squirrelmail-1.2.11.tar.gz | tar xvf -
```

- Cambiar al usuario root. Ir al directorio donde se ha descomprimido las fuentes y copiarlas al sitio donde va a ir el webmail:

```
cd squirrelmail-1.2.11
cp -r * /var/www/webmail
```

- Establecer los permisos adecuados. El root debe ser propietario de todos los ficheros, exceto del directorio `/var/www/webmail/data` que debe ser el usuario y grupo sobre el que se ejecuta Apache. Para ello:

```
cd /var/www/webmail
chown -R root.root *
chown -R apache.apache data
```

12.2. Configuración

El webmail tiene un pequeño programa a través del cual se configura. Para ello en el directorio donde se ha instalado, se ejecuta `./configure`. El menu que sale es el siguiente:

```
SquirrelMail Configuration : Read: config_default.php (1.2.0)
```

```
-----
```

```
Main Menu --
```

1. Organization Preferences
2. Server Settings
3. Folder Defaults
4. General Options
5. Themes
6. Address Books (LDAP)
7. Message of the Day (MOTD)
8. Plugins
9. Database

```
D. Set pre-defined settings for specific IMAP servers
```

```
C. Turn color on
```

```
S Save data
```

```
Q Quit
```

No olvidar una vez realizado todos los cambios, grabarlos

Debe de configurarse:

- El nombre de la organización y lenguaje por defecto. Para ello, opción 1, y en menu que aparece:

```
SquirrelMail Configuration : Read: config_default.php (1.2.0)
```

```
-----
```

```
Organization Preferences
```

1. Organization Name : SquirrelMail
2. Organization Logo : ../images/sm_logo.png
3. Org. Logo Width/Height : (308/111)
4. Organization Title : SquirrelMail \$version
5. Signout Page :
6. Default Language : en_US
7. Top Frame : _top

```
R   Return to Main Menu
C.  Turn color on
S   Save data
Q   Quit
```

Command >>

Camnbiamos *Organization Name* y *Organization Title* a lo que se desee, en nuestro caso Reinos Olvidados, y el *Default Language* a *es_ES*.

- Cambiar la configuración del servidor de correo. Para ello desde el menú principal, opción 2, sale el siguiente menu:

SquirrelMail Configuration : Read: config_default.php (1.2.0)

Server Settings

```
1.  Domain           : mydomain.com
2.  IMAP Server       : localhost
3.  IMAP Port         : 143
4.  Use Sendmail/SMTP : SMTP
6.   SMTP Server      : localhost
7.   SMTP Port        : 25
8.   Authenticated SMTP : false
9.   POP Before SMTP  : false
10. Server            : cyrus
11. Invert Time       : false
12. Delimiter         : detect
```

```
R   Return to Main Menu
C.  Turn color on
S   Save data
Q   Quit
```

Command >>

Debe de cambiarse *Domain* al dominio primario que se esté usando. Debe de cambiarse el tipo de *Server* (opción 11) a *courier*.

13. Administración del sistema

En esta sección se abordan los diferentes aspectos de la gestión de los diversos aspectos del sistema. Todas estas operaciones se van a hacer desde la línea de comandos y como usuario *root*. Las utilidades de *vpopmail* se encuentran situadas en */home/vpopmail/bin*, directorio que normalmente no se encuentra en los caminos de búsqueda del sistema. Debe de hacerse explícitamente referencia al camino completo hasta el fichero (es decir, cuando se habla de *vadduser*, es */home/vpopmail/bin/vadduser*). Por cuestiones de brevedad, en los siguientes apartados se omitirá el camino completo.

13.1. Gestión de dominios

- **Añadir dominio.** Debe de tenerse el nombre del dominio y una clave para el usuario *postmaster* (este debe de existir en todo dominio, y él le llegaran los mails erroneos). Además debe de darse de alta en el DNS el registro MX adecuado apuntado a la IP externa de nuestro sistema. Para añadir un dominio se usará la orden *vadddomain*. La sintaxis es la siguiente:

```
vadddomain dominio clavepostmaster
vadddomain dominio
```

En el segundo caso, el programa preguntará por la clave que se le quiere asignar al *postmaster* (dos veces). Por ejemplo para crear el dominio *dominio.com*

```
vadddomain dominio.com
Please enter password for postmaster:
enter password again:
```

- **Borrar un dominio.** Se usa la orden *vdeldomain*. Tener en cuenta que se borrará el dominio completo existan o no usuarios en el mismo. Por ejemplo, para borrar *dominio.com*:

```
vdeldomain dominio.com
```

- **Añadir un alias.** Permite añadir un nuevo nombre a un dominio, es decir, si tenemos el dominio *dom1*, podemos añadir uno nuevo, *dom2*, de tal manera que sean el mismo. Por ejemplo, queremos que *dominio.es* apunte a *dominio.com*:

```
vaddaliasdomain dominio.es dominio.com
```

13.2. Gestión de usuarios

- **Añadir un usuario.** Se usa la orden *vadduser*. Necesitamos conocer la password y la dirección de correo que queremos. Por ejemplo, para añadir `usuario@dominio.com` con password `clave`:

```
vadduser usuario@dominio.com clave
```

En caso de no especificar la clave, el sistema nos pedirá que se introduzca una.

Opcionalmente se puede especificar un comentario con el parámetro `-c`. Es útil para incluir el nombre del usuario. Ejemplo, crear el usuario llamado Usuario de administración con dirección `adm@dominio.com`:

```
vadduser -c "Usuario de administracion" adm@dominio.com
```

- **Borrar un usuario.** Se usa la orden *vdeluser*. Por ejemplo, para borrar `usuario@dominio.com`:

```
vdeluser usuario@dominio.com
```

- **Obtener la información sobre un usuario.** La orden encargada de ello es *vuserinfo*. Ejemplo, si se quiere obtener información de `adm@dominio.com`:

```
vuserinfo adm@dominio.com
```

La salida obtenida es similar a:

```
name:    adm
passwd:  $1$3z8Ws$5K.AcVNxoPPik7mj8NgFs.
clear passwd: uio
uid:     0
gid:     0
          all services available
dir:     /home/vpopmail/domains/dominio.com/adm
quota:   NOQUOTA
usage:   NOQUOTA
```

- **Cambiar la clave de un usuario.** La orden que se usa es *vpasswd*. Ejemplo, si se quiere cambiar la password de `adm@dominio.com` por `"nuevapassword"`:

```
vpasswd adm@dominio.com nuevapassword
```

En caso de omitir el campo con la nueva clave, el programa preguntará por la misma y pedirá posteriormente confirmación.

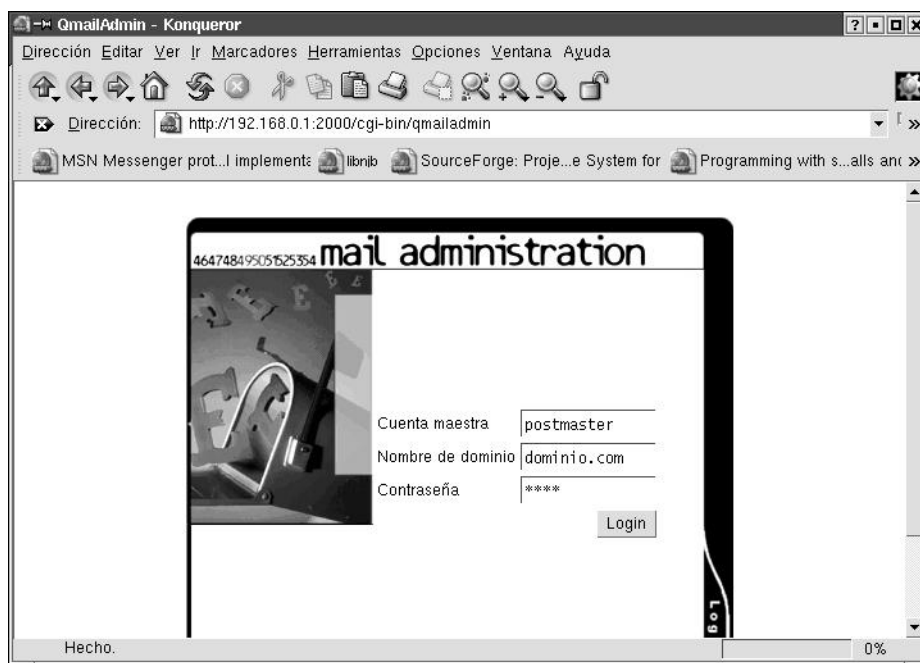
13.3. Herramientas gráficas de administración

Las herramientas descritas en las dos secciones anteriores, permiten el uso de scripts para la gestión. Sin embargo, para dotar de un poco de comodidad al sistema, se ha incluido una gestión web con ayuda de QmailAdmin. Con ayuda de la misma pueden gestionarse las cuentas de cada uno de los dominios.

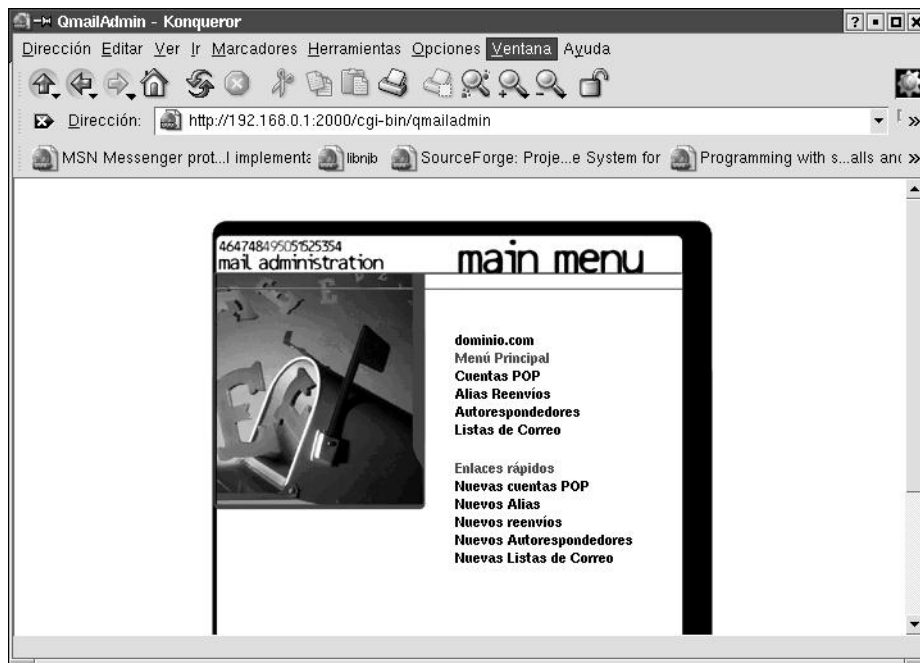
El único inconveniente que tiene es que hay que hacer la gestión dominio a dominio, entrando en el mismo por cada uno del que se quiere gestionar.

En la configuración propuesta a esta herramienta se accede a través de la url <http://192.168.0.1/cgi-bin/qmailadmin>. En principio usa su propia autenticación, basada en la uenta de postmaster del dominio a gestionar. Puesto que no se ha montado esta web de administración con SSL y loas contraseñas viajan en texto claro, hay que usarla bien localmente, bien en una red desde la que se confíe. En otro caso sería conveniente montarla sobre SSL.

Para entrar a gestionar un dominio se presenta la siguiente pantalla



Una vez introducido el dominio y la contraseña de la cuenta *postmaster* se nos presenta el siguiente menú:

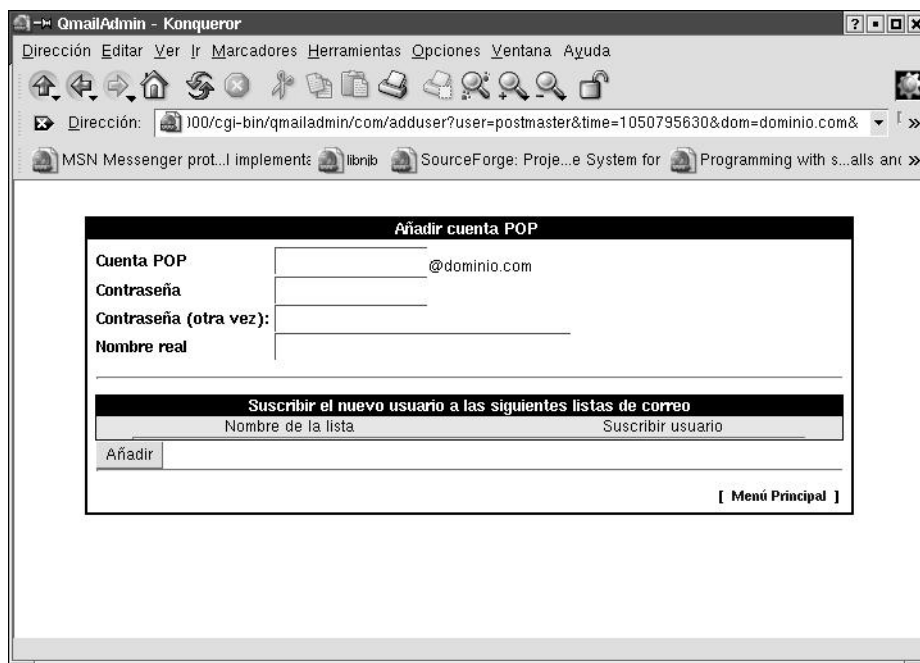


La descripción del las opciones:

- *Cuentas POP*: Toda la gestión de cuentas: añadir usuarios, modificar o borrar. Cambiar las diferentes propiedades de la cuenta como nombre del usuario, clave o subscribirlo a una lista de correo.
- *Alias reenvios*: Permite definir alias a una cuenta de correo.
- *Autoresponderes*: Permite definir respuestas automáticas, aunque es algo a evitar en lo posible, en especial si los usuarios estan subcristos a listas de correo, puesto que son muy molestos (vacation y similares).
- *Listas de correo*: Permite la gestión de listas de correo.

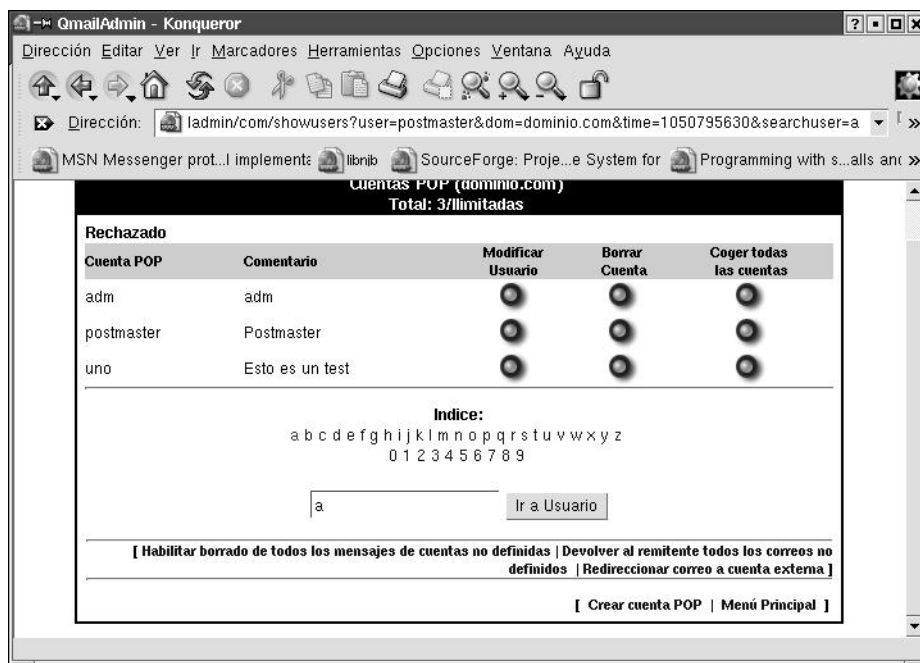
Existen una serie de enlaces rápidos a la creación de nuevas cuentas de correo, alias, autoresponder y listas.

Al pulsar sobre el enlace *Nuevas cuentas POP*, pasando a la siguiente pantalla:



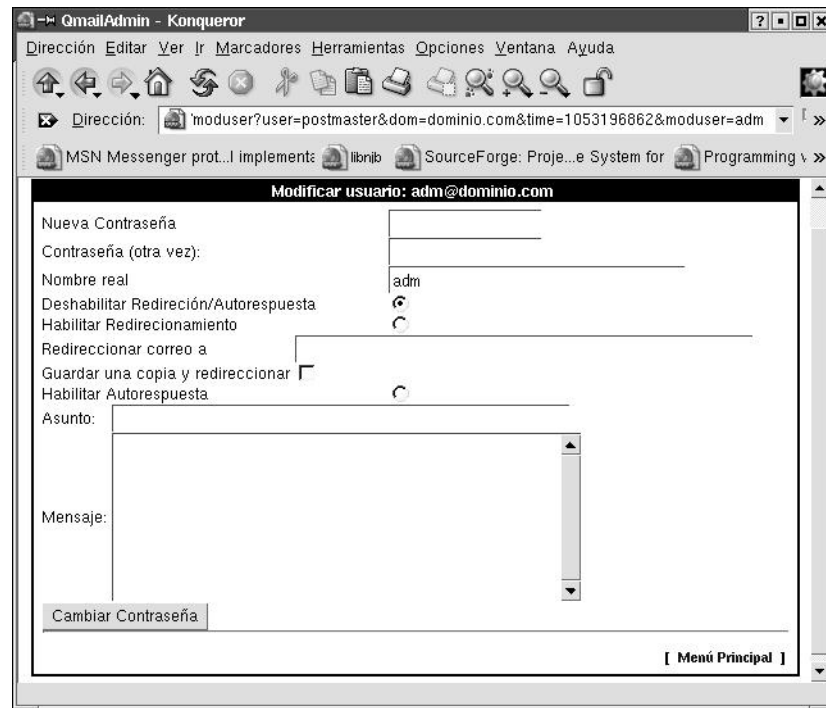
En ella se introduce el nombre de la cuenta que quiere crearse, la contraseña de la misma y el nombre del usuario - campo *gecos* -. Aparece un listado de las listas de correo, donde puede seleccionarse aquellas para las que el usuario deba de estar suscrito.

Si se quiere modificar las propiedades de los usuario se pulsa sobre *Cuentas POP*, pasando a la siguiente pantalla:



Desde esta pantalla se puede:

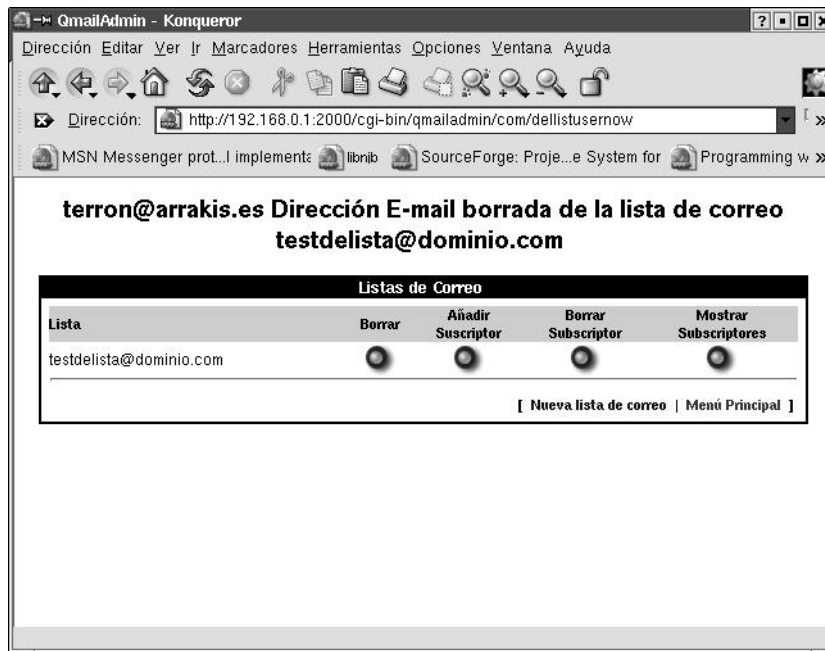
- Borrar una cuenta: para ello se pulsa sobre el botón situado en la columna borrar de la cuenta en cuestión, presentandose una pantalla de confirmación.
- Modificar una cuenta: para ello se pulsa en el botón situado en la columna modificar. Un poco más adelante puede verse la pantalla. Desde ella puede modificarse las claves del usuario, activar el autoresponder - pudiendo especificarse el mensaje -. El único problema que presenta esta herramienta es que hay que modificar también la contraseña para que se almacenen los cambios.
- Habilitar la opción de que todos los correos no definidos sean borrado. Esto se activa al pulsar *Habilitar borrado de todos los mensajes de cuentas no definidas*.
- Habilitar una cuenta como *recoge correos*. La cuenta elegida recibirá todos los correos rebotados. Esta cuenta se selecciona pulsando sobre *Coger todas las cuentas*. Este comportamiento puede deshabilitarse pulsando sobre *Devolver al remitente todos los correos no definidos*.



Para gestionar los alias y reenvios se usa el enlace *Alias reenvio*.

- Alias. Para añadir un alias se pulsa sobre el enlace *Añadir nuevo alias*. Se presenta una lista desplegable, donde se elige la cuenta a la cual va a añadirse el alias. Posteriormente se escribe el alias que queremos usar. También pueden borrarse o modificarse dichos alias.
- Reenvios. Un reenvio pasa el correo que reciba la cuenta del dominio que se administra hacia una cuenta especificada. Se añaden usando la opción *Añadir nuevo reenvio*

El enlace *Listas de correo* del menu principal permite la gestión de las mismas. El menu que se muestra es el siguiente:



Se puede añadir, borrar y ver los suscriptores que tienen una lista. Para ello basta con pulsar sobre el botón en la columna deseada. Existe un botón que permite borrar toda la lista de correos y un enlace que nos permite crear una nueva.

14. Introducción al correo de Internet

A continuación se da una breve introducción a como funciona el correo en Internet, apoyandose en las RFC 2821 - la cual define el protocolo de transporte de correo - y la RFC 2822 - que define el formato de los mensajes.¹¹

En mensaje de correo está compuesto por un sobre (envelope) y un contenido.

El sobre tiene un remitente (sender) y una o más direcciones de destino (recipients). Esta información se utiliza para rutar el correo a su destino y para devolver al remitente los errores.

El contenido no es más que una serie de caracteres del juego US-ASCII (códigos del 1 al 127). Este contenido se encuentra dividido en líneas, es decir conjuntos de caracteres que acaban con CRLF (ASCII 13 y 10). Estas líneas

¹¹En principio solo para mensajes de texto, existiendo una serie de extensiones llamadas MIME para otro tipo de contenido, consultar las RFC 2045, 2046, 2047, 2048 y 2049.

se disponen en una cabecera (header) y un cuerpo (body), el cual es opcional. Estos se encuentran separados por una línea vacía.

La cabecera tiene una estructura especial, con el nombre del campo, ":z el contenido del mismo. No importan las mayúsculas o minúsculas en los nombres de los campos. El formato completo de los mismos puede encontrarse en la RFC 2822.

Los únicos campos obligatorios de la cabecera son la fecha de origen del mensaje (Date:) y los campos de origen (From:)

Entre los campos opcionales importantes de cara al usuario:

- Reply-To: Una lista de direcciones separada por comas que indica a qué buzón(es) (mailbox) debe responderse.
- To: Los destinatarios principales del mensaje.
- Cc: Otros destinatarios no primarios del mensaje (del inglés carbon copy).
- Bcc: Otros destinatarios de los mensajes cuyas direcciones no deben de revelarse (del inglés blind carbon copy).
- Subject: Identifica el asunto del mensaje.
- Message-ID: Un identificador único de mensaje.

Para el correcto rutado y para evitar bucles existen una serie de campos que indican por donde ha pasado el correo. Estos campos deben de ser insertados por el servidor SMTP cuando recibe un correo.

- Return-Path: Este campo indica el buzón al cual debe de mandarse un mensaje en caso de error.
- Received: Campo de traza. Sirve para detectar por donde se ha rutado el correo, bucles o conexiones lentas. Es un campo de depuración, pero importante para la correcta gestión de correo.

14.1. Direcciones

Una dirección de internet es una cadena de caracteres que contiene el signo @. La parte a la derecha del mismo se denomina parte de dominio. La parte a la izquierda se denomina parte local. Como excepción, existe la dirección que es una cadena vacía.

La parte del dominio se identifica un MAIL EXCHANGER (MX) o un host que puede recibir correo y que debe resolverse a través de DNS. Puede indicarse en la parte de dominio una dirección IP entre corchetes, e.j [192.168.1.1]. En esta parte no se distinguen entre mayúsculas y minúsculas.

La parte local se interpreta según el dominio al que pertenezca. Normalmente hace referencia a un buzón en el mismo.

14.2. Protocolo SMTP

El protocolo SMTP es el encargado de transferir el correo entre diferentes puntos de una manera eficiente y rápida. Normalmente usa como transporte una comunicación TCP. El puerto por defecto donde escuchan los servidores es el 25.

En el modelo SMTP tenemos un cliente SMTP y un servidor SMTP. El cliente determina a qué servidor debe de mandar el correo de que dispone - puede ser el servidor final o bien un servidor que le haga de pasarela -.

Una vez establecida la conexión inicial y negociado la capacidades del servidor, se produce una o más transacciones de correo. Estas transacciones siguen un modelo de enviar un comando y esperar una respuesta ¹², lo cual lo hace problematico sobre enlaces que tenga una alta latencia.

Los comandos SMTP esperan una respuesta, donde los 3 primeros caracteres son números, que indican si la acción especificada se ha llevado a cabo con éxito o no.

A continuación se presenta un pequeño esquema de los comandos que aparecen en una transacción de correo.

1. Conexión y negociación de extensiones. La conexión comienza hacia el puerto 25 del servidor, se espera una respuesta positiva (220) puede negociarse las extensiones. Normalmente esto se hace a través del comando EHLO. En caso de que no esté implementado, pasa a usar HELO.
2. Identificación del remitente. El comando MAIL FROM: es el encargado de pasar al servidor el remitente del sobre.
3. Identificación de los destinatarios. El comando encargado es RCPT TO:.
4. Envío de los datos de correo. Esto se hace con DATA. A continuación se van mandando las diferntes líneas que componen el mensaje hasta el final que se marca con CRLF.CRLF

¹²A menos que se use la extesión PIPELINING, RFC 2920

5. Finalizar la transacción con QUIT o bien iniciar un nuevo envío desde el punto 2

En medio de este pequeño esquema pueden producirse errores, se puede resetear la comunicación o puede hacerse otro tipo de consultas que no lleve al envío de correo. Para tener una visión completa debe de consultarse la RFC 2821.

15. Qmail a fondo

En la siguiente sección se explica el funcionamiento completo de qmail, como usa vpopmail para facilitar el uso de los dominios virtuales. Este apartado está estructurado en función del flujo que sigue el correo - ya proceda de la red via SMTP, bien se inyecte localmente en el sistema -, a través de la cola de correo, y de los programas que se encargan de la entrega local o remota.

Todos los ficheros de configuración de los cuales se habla en los siguientes apartados se encuentran en */var/qmail/control*. Sino existe el fichero de configuración, se toma el valor por defecto que se menciona.

La mayoría de la información de este apartado está obtenida de las páginas de manual de qmail. De hecho, la descripción de los ficheros de configuración y las variables de entorno son traducciones de las mismas.

15.1. Entrada de correo via SMTP

El primer elemento es el *tcpserver*. Este programa se encarga de escuchar en el puerto tcp que se le indique. Posteriormente ejecuta el programa que se le indique, de tal manera que la entrada estandar (descriptor 0) del mismo recibe los datos de la conexión de red y la salida estandar (descriptor 1) los envia y asigna ciertos valores a una serie de variables de entorno que el programa llamado puede leer.

Las conexiones pueden permitirse o rechazarse gracias a un fichero de reglas, además de establecer variables de entorno según nos convenga. Con ayuda de este mecanismo se permite el relay a aquellas direcciones que interese. El nombre de la variable es RELAYCLIENT.

Una vez recibida la conexión tcpserver ejecuta el módulo de qmail *qmail-smtpd*. Este programa se encarga de leer el correo a través de SMTP de la red e introducirlo en el sistema (esto lo hace gracias al programa qmail-queue). Antes de inyectar el correo en el sistema cuenta el número de saltos para evitar bucles (esto lo hace a través de las cabeceras Received o Delivered-To).

qmail-smtpd utiliza las siguientes ficheros de configuración (situados en /var/qmail/control):

- *badmailfrom* La lista de direcciones que aparezcan en este fichero son rechazados. Si aparece una linea de tipo @host, rechazará todas las direcciones de dicho host. Por defecto está vacío.
- *databytes* Máximo número de bytes permitidos por el mensaje. Si la variable de entorno DATABYTES esté definida, ignora databytes. Esto puede usarse con ayuda del tcpserver para permitir diversos tamaños de mensaje en función del cliente que haga el relay.
- *localiphost* Si este fichero existe, cuando se recibe un correo con un recipiente del tipo @[a.b.c.d], se sustituirá por el contenido de este fichero. Si no se especifica ninguno se toma por defecto el del fichero me. Esto se hace antes de procesar el rcpthosts.
- *morercpthosts* Sirve para indicar dominios extras que se permiten en el RCPT. En caso de modificar este fichero debe ejecutarse el programa /var/qmail/bin/qmail-newmrh.
- *rcpthosts* Dominios que se permiten en el RCPT. Si este fichero existe, se rechazará cualquier mensaje cuyo dominio no esté en este fichero. No debe arrancarse qmail-smtpd sin la existencia de este fichero porque en ese caso el sistema será permitirá transmitir correo a cualquiera, con los problemas de correo no solicitado¹³. En el caso de que la variable RELAYCLIENT exista, se ignorará este fichero. Esta variable puede definirse con ayuda del tcpserver.
- *smtpgreeting* Mensaje de saludo cuando se conecta un cliente. Por defecto el contenido del fichero me. En caso contrario, la cadena contenida en este fichero. Hacer notar que la primera palabra debería ser el nombre del host donde corre qmail.
- *timeoutsmtpd* Tiempo en segundos durante los cuales esperará por un buffet de datos del cliente remoto. Por defecto o sino existe 1200 segundos.

15.2. Entrada de correo via inyección local

El programa que inyecta el correo local es *qmail-inject* (el envoltorio /var/qmail/bin/sendmail también llama a este programa). Este programa

¹³Conocido en la jerga de Internet como SPAM

lee de la entrada estándar un mensaje, lo preprocesa y lo inyecta en la cola a través de *qmail-queue*.

Además *qmail-inject* se asegura que la cabecera está correctamente construida, añadiendo ciertos campos en caso de que falten y rechazando aquellos mensajes con una cabecera corrupta.

qmail-inject permite el uso de VERP (variable envelope return path). Esta es una extensión de *qmail* para el manejo de listas de correo que permite asignarle un Return-Path: único a cada suscriptor de la lista y así ayudar a un diagnóstico y solución de problemas más rápido, puesto que usando VERP los errores permiten identificar automáticamente la dirección que falla (puesto que está incluida en el Return-Path:) mientras que en caso normal, el mensaje llegaría al administrador de la lista y habría que tratarlo manualmente.

Los ficheros y variables de configuración del programa son los siguientes:

- *defaultdomain* Si no existe este fichero, por defecto toma el valor de *me*. En caso contrario, cualquier nombre de host que no tenga un punto, se le añadirá el contenido de este fichero (excepto aquellos que acaben en + que se le añadirá el contenido de *plusdomains*. Esto también es aplicable a *defaulthost*. Si la variable de entorno *QMAILDEFAULTDOMAIN* existe, su valor tiene preferencia sobre el contenido de este fichero.
- *defaulthost* Nombre de host por defecto. Este valor por defecto es *me*. En caso de que exista el fichero, toma el valor especificado en el mismo. Este valor se añade a cualquier dirección que no presente un nombre de host. Si la variable de entorno *QMAILDEFAULTHOST* existe, su valor tiene precedencia sobre el contenido de este fichero.
- *idhost* Nombre de máquina usado para generar el campo de cabecera Message-Id. Por defecto este valor es el contenido de *me*, tomando el valor especificado en el fichero en caso de que exista. Sin embargo, este fichero debe ser un FQDN dentro del dominio y cada host dentro del mismo debe de tener uno distinto. Si la variable de entorno *QMAILID-HOST* existe, se usa su valor.
- *plusdomain* Este valor se añade a todos los nombres de máquina que acaben con un signo más, incluyendo lo especificado en *defaulthost* (si este acaba en un signo más). Si la variable de entorno *QMAILPLUS-DOMAIN* existe, se usa su valor.
- *QMAILINJECT* Esta variable de entorno controla el comportamiento del programa en función de los valores que tenga:

- **c** Usa para el campo From direcciones con comentarios. Esto indica que en el campo From el nombre del destinatario estará entre paréntesis seguido de la dirección sin `¡¿`, por ejemplo (Carlos) `terron@arrakis.es`, frente al uso más común de `Carlos ¡terron@arrakis.es¡`
- **s** No use el campo Return-Path. Si no se especifica esta opción y en el mensaje que se le pasa a `qmail-inject` existe dicho campo, se usará para la dirección del remitente, tomando precedencia sobre todas las variables de entorno. De todas maneras, `qmail-inject` borra dicho campo del mensaje.
- **f** Normalmente `qmail-inject` usa el campo From del mensaje - en caso de que no exista, genera uno -. Con esta opción, borrará el campo From del mensaje y generará uno.
- **i** Borra el campo Message-ID si se le pasa uno, generando uno nuevo.
- **r** Usa un VERP por destinatario. `return path`).
- **m** Usa un VERP por mensaje.

Las opciones en que admite en las línea de comandos son las siguientes:

- **-a** Manda el mensaje a los destinatarios que se indican en la línea de comando ignorando la información de cabecera que contenga el mensaje.
- **-h** Manda el mensaje a los destinatarios que están en la cabecera del mismo.
- **-A** Opción por defecto. Manda el mensaje a los destinatarios que se le pasen en la línea de comandos, y en caso de que no existan, los que hay en la cabecera del mensaje.
- **-H** Manda el mensaje a los destinatarios que se le pase en la línea de comandos y a los que se indican en la cabecera del mensaje.
- **-f** Pasa como remitente la dirección que se le indique, ignorando el campo Return-Path: si existe en la cabecera del mensaje y todas las variables de entorno.
- **-N** Opción por defecto, pasa el mensaje a `qmail-queue`.
- **-n** Imprime el mensaje en vez de pasarse a `qmail-queue`.

15.3. Encolado de correo

El programa encargado de meter un correo en la cola de salida es *qmail-queue*. Este programa lee el mensaje y la información del sobre (remitente y destinatario) y coloca el mensaje en la cola de correo para su posterior envío a través de *qmail-send*.

Aparte de esto, añade una línea de traza consistente en un campo Received.

15.4. Gestión de la cola de salida

El programa encargado de la misma es *qmail-send*. Este programa llama a *qmail-lspawn* si tiene que hacer una entrega de correo local, mientras que llama a *qmail-rspawn* si la entrega de correo es remota.

qmail-send puede recibir señales para termine (TERM) o bien para que intente mandar inmediatamente todo los mensajes de la cola (ALRM). Las señales pueden mandarse con ayuda del comando kill. Si recibe la señal HUP relea los ficheros de configuración locals y virtualdomains - para que relea el resto hay que parar y arrancar -.

qmail-send está conectado a *qmail-lspawn*, *qmail-rspawn* y *qmail-clean* a través de pipes, mandado los comandos adecuados a estos programas.

Los diferentes ficheros de configuración que afectan a *qmail-send* son los siguientes:

- *bouncefrom* El usuario desde donde salen los "bounces." mensajes devueltos erróneamente. Por defecto MAILER-DAEMON
- *bouncehost* El host origen de los bounces. Por defecto me. Normalmente un mensaje rebotado tendrá el campo from como bouncefrom@bouncehost. Cuando se envía un mensaje de este tipo el remitente del sobre es una cadena vacía (si se mira el campo Return-Path: de un mensaje de este tipo es ¡!).
- *concurrencylocal* Número máximo de entregas de correo local simultáneas. Por defecto es 10 y puede aumentarse hasta 120.
- *concurrencyremote* Número máximo de entregas de correo remoto simultáneas. Por defecto es 10 y puede aumentarse hasta 120. Este parámetro es importante controlarlo, puesto que puede producirse un cuello de botella a la hora de la entrega remota de correo. Si se hace cero, para la entrega remota de correo.

- *doublebouncehost* Por defecto me, host al que se manda los double bouncer, es decir, aquellos mensajes de error que no han podido entregarse.
- *doublebounceto* Usuario que recibe los double bounces, por defecto postmaster.
- *envnoathost* Por defecto me, en caso contrario el contenido del fichero. Indica el dominio que se le añade a una dirección que no tenga "@". Es decir, si pasa una dirección que es terron y el contenido del fichero es arrakis.es, la dirección quedará reescrita como terron@arrakis.es
- *locals* Lista de dominios a los cuales se hace entrega local, un por linea. Por defecto me.
- *percenhack* Dominios donde el truco de reescritura de % se aplica. Cualquier dominio listado en este fichero hace que una dirección del tipo user%fqn@dominio sea reescrita como user@fqn. A su vez user puede tener % con lo cual se puede volver a aplicar. Este proceso se lleva a cabo antes de procesar locals.
- *queuelifetime* Número de segundos que un mensaje puede estar en la cola. Por defecto es una semana (604800).
- *virtualdomains* Lista de usuarios o dominios virtuales. Un usuario virtuales es de la forma user@dominio:prefijo. Cuando el qmail-send ve que el destinatario de un mensaje es user@dominio, lo reescribirá como prefijo-user@dominio y lo tratará como si fuera local.

En el caso de un dominio virtual la linea sería dominio:prefijo. Cualquier destinatario de ese dominio sería reescrito como prefijo-destinatario@dominio y el mensaje sería tratado como si fuera local.

Se permiten caracteres comodines a la hora de definir los dominios virtuales, por ejemplo todo el mail que fuera a .loquesea.org incluyendo subdominios sería .loquesea.org:john, o incluso dejarlo vacio para que cogiera todo el correo con una linea del tipo :prefijo.

Puesto que posteriormente se iniciará una entrega local de correo, este mecanismo permite usar las extensiones a las direcciones de qmail para gestionar los dominios virtuales.

Vpopmail modificará el fichero virtualdomains con los parámetros adecuados cuando se añade o borra un dominio.

Cada cierto tiempo, qmail-send manda comandos a *qmail-clean* para limpiar la cola de correo.

15.5. Entrega de correo local

La entrega de correo local comienza cuando *qmail-send* da instrucciones a *qmail-lspawn*.

Este proceso se arranca con un modo de entrega determinado (bien a un buzón en formato mbox tradicional, bien un maildir, parámetro que pasará a *qmail-local*.

Se busca los usuarios que controlan las diferentes direcciones. Para ello se sigue el siguiente orden:

1. Mecanismo *qmail-users*. Se procesa el fichero */var/qmail/users/asigns*. En caso de que la dirección no esté gestionada por este mecanismo
2. Se llama a *qmail-getpw*, que se encarga de buscar la cuenta en los mecanismos normales del sistema operativo, fundamentalmente usando la llamada *getpwnam*.

Una vez obtenido esta información y chequeado una serie de parámetros de seguridad, *qmail-lspawn* llamará a *qmail-local* para hacer la entrega.

qmail-local añade un nuevo campo a la cabecera que es *Delivered-To*: indicando el destinatario del mensaje. Si ese mismo campo ya aparece en la cabecera, el mensaje rebotará (bounce) como error, puesto que se ha detectado un bucle - siempre que la entrega sea a un buzón local -.

También añade un nuevo campo *Return-Path*: con el remitente del mensaje.

15.5.1. Extensiones de direcciones

La particularidad más importante que tiene *qmail* respecto a otros MTA es las extensiones de las direcciones. Cualquier usuario local, *user*, es capaz de controlar las direcciones *user-loquesea*. El comportamiento para la dirección *user-loquesea* es controlado por el fichero situado en el directorio *\$HOME*¹⁴ del usuario *user*, por el fichero *.qmail-loquesea*. Es decir, un correo cuyo destinatario sea *john-seguridad@dominio.com*, si *john* es un usuario local del sistema, *qmail* seguirá las instrucciones contenidas en el fichero *\$HOME/.qmail-seguridad*.

Un detalle de este mecanismo es que si contiene *."* la extensión se convierte en *":*", para evitar que se pueda navegar por el árbol de directorios a la hora de leer los ficheros.

El contenido del fichero *.qmail-extension* permite controlar la entrega del correo. El formato de dicho fichero es el siguiente:

¹⁴En los shells de Unix esta variable contiene el directorio raíz del usuario

- Comentarios: un `#` al principio de la línea.
- Un programa al que se le pasa el mensaje. Para ello se utiliza el símbolo del pipe `"—"` seguido del comando que se quiere ejecutar. En este caso *qmail-local* lo ejecutará como `sh -c` comando, enviando a la entrada estándar del mismo el mensaje. En este caso no se añaden las cabeceras *Delivered-To:* o *Return-Path:*. Además se establecen una serie de variables de entorno, entre otras *SENDER* con la dirección del remitente, *RECIPIENT* con la dirección del destinatario, *LOCAL* la parte local de la dirección, *HOST* la parte de dominio, etc. La lista completa de variables se encuentra en la página de manual *qmail-command*.
- Un forward. Hace que el mensaje sea reenviado a la dirección que se especifique después de un `"&"`. Por ejemplo, si el usuario jeff quiere reenviar todo su correo a `jeff@otrodominio.com`, pondría una línea en su `$HOME/.qmail` que sería `&jeff@otrodominio.com`.
- Un buzón tipo *mbox*. Para ello una línea que comienza con `/.º` y que no termina en `/` indicando el nombre del buzón. Por ejemplo que todo el correo vaya a `/home/jeff/buzon`, en `$HOME/.qmail` sería: `/home/jeff/buzon`.
- Un buzón de tipo *Maildir*. Exactamente igual que el anterior solo que la ruta acaba en el buzón *Maildir* (creado con *maildirmake*), acabe en `/`. Si el buzón está en el directorio `$HOME/Maildir`, el contenido sería `./Maildir/` - puesto que *qmail-local* se ejecuta en el directorio `$HOME` del usuario al que le entrega el correo -.

A la hora de procesar las direcciones *qmail*, sigue el siguiente orden:

1. Si la dirección no tiene extensión, es decir es el usuario a secas, y no existe el fichero `$HOME/.qmail`, simplemente lo trata como si estuviera vacío, entregando el correo por defecto.
2. Si existen extensiones en la dirección de entrega:
 - a) Si existe *.qmail-extension* se siguen las instrucciones del fichero.
 - b) Si no existe lo anterior, se quita una extensión y se intenta *.qmail-extension.recortada-default*, así hasta que se llega a *.qmail-default*, que en caso de que no exista hace que se produzca un bounce con el mensaje. Así si la dirección es `terron-seguridad-microsoft@arrakis.es`, los ficheros serían *.qmail-seguridad-microsoft*, *.qmail-seguridad-default* y *.qmail-default*.

En caso de que un mensaje sea reenviado desde un fichero *.qmail-extension* puede controlarse la dirección del remitente y además generar VERP.

Así si un mensaje es entregado a `user-extension@dominio.com`, gestionado por `$HOME/.qmail-extension` (o bien el está en el `$HOME/.qmail-default`), si existe el fichero `$HOME/.qmail-user-extension-owner`, el remitente del mensaje cambiará a `user-extension-owner@dominio.com`.

Si lo que se desea es crear un VERP a cada mensaje que se reenvia, se deben de crear los ficheros `$HOME/.qmail-user-extension-owner` y `$HOME/.qmail-user-extension-owner-default`, esto cambiará el remitente del mensaje a `user-extension-@dominio-@[]`.

Estas extensiones de direcciones permiten la fácil creación de listas de correo por parte del usuario sin necesidad de molestar a los administradores de sistema¹⁵.

15.5.2. Alias locales

Hasta ahora, se ha hablado de direcciones asociadas a usuarios locales. ¿Pero que ocurre con los alias tradicionales?. *qmail* resuelve este problema con ayuda de un usuario alias y de ficheros *.qmail* creado en dicho usuario. Por defecto estos alias se sitúan en `/var/qmail/alias`. Así se crean los alias para `root` - recordemos que *qmail* nunca entrega correo al usuario `uid = 0` -, `mailer-daemon` y `postmaster`.

15.6. Entrega de correo remoto

El módulo *qmail-rspawn* es el encargado de recibir desde *qmail-send* las instrucciones para la entrega remota. Posteriormente ejecuta *qmail-remote* de manera asíncrona para efectar las entregas. Todas estas instrucciones las recibe a través de pipes.

La línea de comandos con la cual se llama *qmail-remote* es:

```
qmail-remote host sender recip [ recip ... ]
```

Donde *host* es el MX o la máquina a la cual se le envía el correo, *sender* es la dirección del remitente, y *recip* son los destinatarios. *qmail-remote* impone una serie de restricciones a los mensajes para que sean válidos según el protocolo SMTP: hace terminar las líneas con CRLF y duplica los "." que están han principio de línea. Es decir, si una línea empieza por ".", se convierte en "..".

Los ficheros de configuración que utiliza *qmail-remote* son los siguientes:

¹⁵De hecho, el paquete *ezlml*, del mismo autor de *qmail* usa todo estos mecanismos

- *helohost* cadena que se manda a un host remoto en el EHLO o HELO. Por defecto me, en caso contrario el contenido de este fichero.
- *smtproutes* este fichero permite controlar las rutas que va a seguir el correo en función de los dominios. Su contenido son líneas tal que:

`dominio:relay`

El dominio indica hacia donde va el correo. El relay es la máquina hacia donde ese dominio debe rutarse. Opcionalmente *relay* puede contener un puerto determinado, separado por dos puntos.

A la hora de indicar el dominio se puede usar comodines. Por ejemplo .us.es indica todas las máquinas que están por debajo del dominio us.es sin incluir éste. Si dominio está vacío (es decir es una línea del tipo :relay), significa que todo el correo se rutará por relay, mientras que si lo que está vacío es relay, significa que ese dominio sigue el mecanismo normal de MX.

- *timeoutconnect* Número de segundos que esperará para conectar con un sistema remoto. Por defecto es 60. El kernel del sistema en el que se ejecute qmail afecta sobre el máximo valor que puede tomar este parámetro.
- *timeoutremote* Número de segundos que se esperará a las respuestas del servidor remoto. Por defecto 60. Al igual que el caso anterior, este parámetro puede verse influido por el kernel sobre el cual se ejecuta qmail.

15.7. Asignación de direcciones de correo a usuarios

Este mecanismo permite asignar varias direcciones a un mismo usuario. Vpopmail hace uso de él para implementar toda la gestión de dominios virtuales con un sólo usuario.

La asignación de direcciones se realiza en el fichero `/var/qmail/users/assign`. Cada una de las asignaciones ocupa una línea y el fichero acaba con un `”.”`.

Tenemos dos tipos de asignación:

- Asignación simple. En este caso no se permiten direcciones comodín. Una asignación simple tiene la forma:

`=local:user:uid:gid:homedir:dash:ext:`

Así *local* controla la dirección local que se trata, *user* el usuario local de la máquina, *uid* el user id, *gid* el group id, *homedir* el directorio casa del usuario¹⁶, *dash* (") y *ext* sirven para hacer referencia al fichero *.qmail-ext* que en *homedir* controlará la dirección.

- Asignación con comodines. Cada línea de extensión de comodines tiene la siguiente forma:

```
+loc:user:uid:gid:homedir:dash:pre:
```

Una línea de este tipo será válida para cualquier dirección que comience por *loc*. El resto de los campos son iguales al caso anterior.

Existe una prioridad en cuanto a las asignaciones, teniendo más precedencia una asignación simple, y posteriormente aquella que sea más específica en caso de estar usando asignaciones con comodines.

Esta información se compila en un fichero especial para acceder a ella lo más rápidamente posible. Para ello se ejecuta el programa *qmail-newu*.

15.8. qmail y vpopmail

En este apartado se comenta cual es el proceso que sigue vpopmail a la hora de añadir dominios y usuarios al sistema de correo y como se produce esa integración con qmail.

Cuando se añade un dominio nuevo, se modifica *rcpthosts*, añadiendo el nuevo dominio del que va a ser uso, se añade una línea a *virtualdomains* de tal manera que el prefijo que se usará será el mismo nombre de dominio.

Por último se modifica el fichero */var/qmail/users/assign* con una línea de tipo comodín que hará que todo el correo hacia dominio.com, sea entregado hacia el usuario con el cual se compiló vpopmail. Posteriormente, para cada usuario que crea, aparte de darlo de alta en la base de datos - para su uso con pop e imap - , simplemente crea un fichero *.qmail-user* en el directorio especificado en la línea de assign, y bajo ese directorio se crea los buzones Maildir de cada usuario.

Cuando se crea un nuevo dominio siempre se crea el usuario postmaster, necesario según las RFC de correo electrónico.

Un fichero *.qmail-default* se crea para cazar todos los usuarios que no existen y devolver el mensaje de error correspondiente (en este fichero lo que se hace es llamar al programa *vdelivermail*). con el cual se compiló vpopmail. Posteriormente, para cada usuario que crea, aparte de darlo de alta en la

¹⁶realmente no tiene porqué ser así

base de datos - para su uso con pop e imap - , simplemente crea un fichero *.qmail-user* en el directorio especificado en la linea de assign.

Un fichero *.qmail-default* se crea para cazar todos los usuarios que no existen y devolver el mensaje de error correspondiente (en este fichero lo que se hace es llamar al programa *vdelivermail*).