Cortafuegos COMO

David Rudder, drig@execpc.com
Traducido por Carlos García Arques cgarcia@isabel.dit.upm.es Traducción v0.2, 13 de Agosto de 1996

El objetivo de este documento es enseñar las bases de la instalación de un cortafuegos mediante un PC y Linux. También trata la instalación y el uso de Servidores Proxy, dispositivos estos con los que se consigue un mayor nivel de acceso a la Internet desde detrás de un cortafuegos. Titulo original: "Firewalling and Proxy Server HOWTO" Título alternativo propuesto por el autor: "The ride of the lonely local area network", es decir "La balada de la red local solitaria".

Índice General

1 Introducción

Los cortafuegos han adquirido gran popularidad de un tiempo a esta parte como el último grito en seguridad en la Internet. Como la mayoría de las cosas que la adquieren, con la popularidad han llegado los malentendidos. En este *Howto* se cubrirán las bases de lo que es un **cortafuegos**, cómo configurar uno, qué **servidores proxy** hay, cómo configurarlos, y las aplicaciones de esta tecnología fuera del campo de la seguridad.

1.1 Realimentación

Todo apoyo o crítica a este documento será bienvenido. Estoy buscando con especial interés críticas de la gente que usa Macintoshes, ya que la información que tengo de ellos es escasa. **¡¡¡POR FAVOR, COMUNICADME CUALQUIER INEXACTITUD EN ESTE DOCUMENTO!!!** Soy humano, y puedo cometer errores. Si encontráis alguno, me interesará mucho conocerlo. Intentaré contestar a todo el correo, pero estoy ocupado; así que, que nadie se ofenda si no lo hago.

Mi dirección de correo electrónico es `drig@execpc.com`

1.2 Renuncia de Responsabilidad

Este documento intenta ser una introducción al funcionamiento de los cortafuegos y servidores proxy. No soy, ni pretendo ser, un experto en seguridad. Soy simplemente un tipo que ha leído mucho y al que le gustan los ordenadores más que al resto de la gente. NO SOY RESPONSABLE DE NINGÚN DAÑO

PRODUCIDO POR ACCIONES CON BASE EN ESTE DOCUMENTO. Por favor, escribo esto para ayudar a la gente a entender el tema, no estoy preparado para hacer depender mi vida de la exactitud de lo que hay aquí.

[1]

1.3 Propiedad Intelectual

1.4 Las Razones para Escribir Esto

Hubo muchos artículos en comp.os.linux.* a lo largo de, más o menos, el año pasado pidiendo ayuda sobre cortafuegos. Parecía como si nadie fuera a contestarlos. Supuse que la razón era que nadie sabía cómo. Así que dediqué cierto tiempo a jugar con cortafuegos y a aprender. Este documento existe como respuesta a aquellas peticiones.

1.5 Por Hacer

- Aprender cómo hacer esto en un Macintosh
- Aprender sobre diferentes paquetes de TCP/IP para Güindous
- Encontrar un buen servidor proxy de UDP que funcione con LiNUX

1.6 Lecturas de Interés

- La documentación del juego de herramientas de TIS
- El Howto del NET-2
- PPP-Como
- El Howto de la Ethernet
- El MINIHowto de las Múltiples Ethernets

[1]**Nota del Traductor:** y yo menos.
(es decir, suscribo el párrafo anterior, excepto donde dice escribir, que debe leerse traducir. Y donde dice experto en seguridad léase traductor experto).

- Networking with LiNUX
- La guía del administrador de red TCP/IP de O'Reilly and Associates

Las herramientas para cortafuegos de TIS traen una colección de documentos que se encuentran entre los mejores que he leído sobre cortafuegos y asuntos relacionados. En la sección 4 se habla más de las *herramientas de TIS*.

2 Cortafuegos: Conceptos Básicos

Cortafuegos es el término que se emplea para referirse a una franja de bosque que se limpia de árboles, vegetación, y cualquier materia inflamable, con el fin de crear una barrera que el fuego de un posible incendio no sea capaz de atravesar.

Un **cortafuegos** en el mundillo de las redes de ordenadores es un dispositivo lógico que protege una red privada del resto de la red (pública). Funcionan así:

1. Se toma un ordenador con capacidad de rutar (por ejemplo un PC con LiNUX)
2. Se le ponen dos interfaces (por ejemplo interfaces serie, o ethernet, o de paso de testigo en anillo (Token Ring), etc...)
3. Se le deshabilita el reenvío de paquetes IP (IP forwarding)
4. Se conecta una interfaz a la Internet
5. Se conecta la otra interfaz a la red que se quiere proteger

Ahora hay dos redes distintas que comparten un ordenador. El ordenador cortafuegos, al que de ahora en adelante llamaremos `"cortafuegos"`, puede comunicarse tanto con la red protegida como con la Internet. La red protegida no puede comunicarse con la Internet, y la Internet no puede comunicarse con la red protegida, dado que hemos deshabilitado el reenvío IP en el único ordenador que las conecta.

Si se quiere llegar a la Internet desde la red protegida, hay que hacer primero un telnet al cortafuegos, y acceder a la Internet desde él. Del mismo modo, para acceder a la red protegida desde la Internet, se debe antes pasar por el cortafuegos.

Este es un mecanismo de seguridad excelente contra ataques desde la Internet. Si alguien quiere atacar la red protegida, primero tiene que atravesar el cortafuegos. De esta manera el ataque se divide en dos pasos, y, por lo tanto, se dificulta. Si alguien quiere atacar la red protegida por métodos más comunes, como el bombardeo de emails, o el nefasto `"Gusano de Internet"`, simplemente no podrá alcanzarla. Con esto se consigue una protección excelente.

2.1 Inconvenientes de los Cortafuegos

El mayor problema de los cortafuegos es que restringen mucho el acceso a la Internet desde la red protegida. Básicamente, reducen el uso de la Internet al que se podría hacer desde un terminal. Tener que entrar en el `cortafuegos` y desde allí realizar todo el acceso a Internet es una restricción muy seria. Programas como *Netscape* (pronúnciese Nescafé), que requieren una conexión directa con la Internet, no funcionan desde detrás de un cortafuegos. La solución a todos estos problemas es un **Servidor Proxy**.

2.2 Servidores Proxy

Los servidores proxy son un invento que permite el acceso directo a la Internet desde detrás de un cortafuegos. Funcionan abriendo un socket en el servidor y permitiendo la comunicación con la Internet a través de él. Por ejemplo: si mi ordenador, `drig`, estuviera dentro de la red protegida y quisiera ver el Web con *Netscape*, pondría un servidor proxy en el `cortafuegos`. El servidor proxy estaría configurado para hacer que las peticiones de conexión de mi ordenador al puerto 80 de otra máquina, se conectara a su puerto 1080, y él mismo establecería una conexión con el puerto 80 de la máquina deseada. A partir de entonces reenviaría todos los datos de esa conexión a la otra máquina.

Quien haya usado *TIA* o *TERM* se ha encontrado este concepto antes. Con estos dos programas se puede redirigir un puerto. Un amigo tenía TIA configurado para hacer que quien se conectara a la 192.251.139.21 puerto 4024 lo hiciera a su servidor de Web. El servidor proxy funciona así pero al revés. Para conectarnos al puerto 80 de cualquiera, debemos usar el puerto 1080 (o cualquier otro que hayamos dispuesto) del servidor proxy.

Lo importante de los servidores proxy es que, bien configurados, son completamente seguros. *No* dejan que nadie entre a través de ellos.

3 Configuración

3.1 Requerimientos de Hardware

Para nuestro ejemplo, el ordenador es un 486-DX66 con 8 Megas de RAM, una partición para Linux de 500 Megas, y una conexión PPP a un proveedor de Internet a través de un módem de 14.400 bps . Ese es nuestro linux básico. Para convertirlo en un cortafuegos le añadimos una tarjeta Ethernet NE2000. Con ella queda conectado a tres PC'es con Güindous 3.1 y Trumpet Winsock, y a dos Sun'es con SunOs. La razón de elegir este escenario es que son las dos plataformas con las que estoy familiarizado. Imagino que gran parte de lo que cuento aquí es factible con Mac'es pero, dado que no uso Mac'es con suficiente asiduidad, lo cierto es que no lo sé.

3.2 Configurando el Software

Así, que ahora tenemos un LiNUX conectado a Internet por una linea PPP de 14.400 . Además tenemos una red Ethernet que conecta el LiNUX y el resto de los ordenadores. Lo primero, debemos recompilar el núcleo con las opciones apropiadas. En este momento yo echaría un vistazo al *Kernel-Como*, al *Ethernet-HOWTO*, y al *NET-3-HOWTO*. Luego teclearía `"make config"`:

Las opciones requeridas son:

1. Habilitar el Soporte de Red
2. Habilitar la opción de red TCP/IP (TCP/IP Networking)
3. Deshabilitar el reenvio de paquetes IP (CONFIG_IP_FORWARD)
4. Habilitar la opción de Cortafuegos IP (IP Firewalling)
5. Probablemente, habilitar las cuentas IP (IP Accounting). Parece razonable, dado que estamos configurando un dispositivo de seguridad
6. Habilitar el Soporte de Dispositivos de Red (Networking Device Support)
7. Habilitar el soporte de PPP y Ethernet, aunque esto depende del tipo de interfaces que se tenga en cada caso

Ahora, recompilamos y reinstalamos el núcleo y rearrancamos la máquina. Las interfaces deberían ser reconocidas en la secuencia de arranque para que todo estuviera bien. Si no, habría que repasar los Howtos antes mencionados y volverlo a intentar hasta que funcionase.

3.3 Las Direcciones de Red

Esta es la parte interesante. Dado que no queremos que la Internet tenga acceso a nuestras máquinas, no necesitamos usar direcciones reales. Una buena elección es el rango de direcciones de clase C 192.168.2.xxx, que está designado como rango para pruebas. Es decir, nadie lo usa, y no entrará en conflicto con ninguna petición al exterior. De modo que, en esta configuración, sólo se necesita una dirección IP real. Las otras se pueden elegir libremente y de ninguna manera afectarán a la red.

Asignamos la dirección IP real al puerto serie del `cortafuegos` que usamos para la conexión PPP. Asignamos 192.168.2.1 a la tarjeta Ethernet del `cortafuegos`. Asignamos a las otras máquinas de la red protegida cualquier dirección del rango anterior.

3.4 Pruebas

Lo primero es hacer ping a la internet desde el `cortafuegos`. Yo antes usaba `nic.ddn.mil` como punto de prueba. No deja de ser una buen sitio, pero ha demostrado ser menos fiable de lo que esperaba. Si no funciona a la primera, probaremos a hacer pings a otro par de sitios que no estén conectados a nuestra red local. Si no funciona es que el PPP está mal configurado. Tendríamos que volver a leer el *Net-3-HOWTO* y a probar.

Ahora probaremos a hacer pings entre las máquinas de la red protegida. Todas deben ser capaces de hacer ping a las demás. Si no fuera así, habría que leer de nuevo el *Net-3-HOWTO* y trabajar en la red un poco más.

Ahora, todas las máquinas de la red protegida deben ser capaces de hacer pings al `cortafuegos`. Si no, vuelta atrás. Recuerda: deberían ser capaces de hacer ping a la 192.168.2.1, no a la dirección PPP.

Entonces probaremos a hacer ping a la dirección PPP del `cortafuegos` desde dentro de la red protegida. *No debe funcionar*. Si funciona es que no hemos deshabilitado el *Reenvio del Paquetes IP* y habrá que recompilar el núcleo. Al haber asignado a la red protegida la dirección 192.168.2.0 ningún paquete será encaminado a ella por la Internet, pero, en cualquier caso, es más seguro tener el reenvío de paquetes IP deshabilitado. Esto deja el control en nuestras manos, no en las manos de nuestro proveedor de PPP.

Finalmente, haremos ping a todas las máquinas de la red protegida desde el `cortafuegos`. Llegados a este punto, no debería haber problemas.

Ya tenemos una disposición de cortafuegos básica.

3.5 Seguridad para el Cortafuegos

El cortafuegos no sirve si lo dejamos vulnerable a los ataques. Primero echaremos un vistazo al /etc/inetd.conf. Este es el fichero de configuración del así llamado "superservidor" (inetd), que arranca un buen número de demonios servidores cuando les llega una petición.

Entre ellos:

- Telnet
- Talk
- FTP
- Daytime

Se debe desactivar todo lo que no se necesite. No dudaremos en desactivar netstat, systat, tftp, bootp, y finger. Seguramente querremos desactivar telnet, y dejar sólo rlogin, o viceversa.

Para desactivar un servicio basta con poner un # al comienzo de la linea que se refiera a él. Después hay que mandar una señal SIG-HUP al proceso inetd tecleando "`kill -HUP <pid>`", donde `<pid>` es el número de proceso de inetd. Esto hará que inetd relea su fichero de configuración (`inetd.conf`) y se reinicie. Lo comprobaremos haciendo un telnet al puerto 15 del `cortafuegos`, el puerto de netstat. Si aparece la respuesta de netstatd, no hemos reiniciado inetd correctamente.

4 Software para Cortafuegos

4.1 Paquetes disponibles

Un cortafuegos en sentido estricto no necesita ningún software aparte del núcleo de LiNUX y los programas básicos de red (inetd, telnetd y telnet, ftpd y ftp). Pero un cortafuegos así es extremadamente restrictivo y no muy útil.

Así que la gente ha hecho programas para aumentar la utilidad de los cortafuegos. El que examinaremos con mayor detalle es un paquete llamado "*socks*", que implementa un `servidor proxy`. Sin embargo, existe otro par de programas que hay que tomar en consideración. Ahora les daremos un rápido repaso.

4.2 El juego de herramientas para cortafuegos de TIS

TIS ha sacado una colección de programas para facilitar la realización de cortafuegos. Básicamente, los programas hacen lo mismo que el paquete *Socks*, pero tiene una estrategia de diseño diferente. Mientras que `Socks` tiene un único programa que cubre todas las operaciones de la Internet, `TIS` provee un programa para cada utilidad que quiera usar el cortafuegos.

Para compararlos mejor, veamos el ejemplo del acceso al *Web* y por *Telnet*. Con `Socks`, hay que hacer un fichero de configuración y poner en marcha un demonio. Mediante ese fichero y ese demonio se activan tanto el `Telnet` como el `Web`, así como cualquier otro servicio que no se haya desactivado explícitamente.

Con las `herramientas TIS`, se arranca un demonio para el `Web` y otro para el `Telnet`, y se escribe un fichero de configuración para cada uno. Después de haber hecho eso, el resto de formas de acceso a Internet siguen prohibidas hasta que se configuren explícitamente. Si no existe un demonio especial para una determinada utilidad (por ejemplo, para talk), hay un demonio *"para todo"* pero no es ni tan flexible, ni tan fácil de configurar como las otras herramientas.

Esto puede parecer una diferencia menor, pero en realidad es una gran diferencia. `Socks` permite ser desidioso. Con un servidor de `Socks` mal configurado la gente de dentro tiene más acceso a la Internet del que se quería. Con las herramientas `TIS`, la gente del interior tiene solamente el acceso que el administrador del sistema quiera que tengan.

`Socks` es más fácil de configurar, más fácil de compilar, y permite una mayor flexibilidad. El `juego de herramientas de TIS` es mas seguro si se quiere controlar a los usuarios *de dentro*. **Los dos proporcionan una protección absoluta del exterior**.

4.3 El Limitador de TCP (TCP Wrapper)

El limitador de TCP no es una utilidad de cortafuegos, pero sirve para algo parecido. Usando el limitador de TCP podemos controlar quién tiene acceso a nuestra máquina y a qué servicios, así como registrar las conexiones. También ofrece detección básica de impostores.

El limitador de TCP no se cubre de manera más extensa aquí por un par de razones:

- No es un verdadero cortafuegos.
- Para utilizarlo se tiene que estar directamente conectado a la Internet, es decir, se tiene que tener una dirección IP.
- Sólo controla la máquina en la que está instalado, y por lo tanto no sirve para una red. Los cortafuegos pueden proteger todas las máquinas cualquiera que sea su arquitectura. El limitador de TCP no funciona en Macintoshes ni en MS Güindouses.

5 Instalación del Servidor Proxy

El servidor proxy requiere software adicional. Éste se puede conseguir en:

`ftp://sunsite.unc.edu/pub/LiNUX/system/Network/misc/socks-Linux-src.tgz`

Solamente hay un ejemplo de fichero de configuración en ese directorio, se llama `"socks-conf"`. Debemos descomprimir y desempaquetar los ficheros en un directorio de nuestro ordenador, y seguir las instrucciones de cómo compilarlo. Yo tuve un par de problemas compilándolo. Hay que asegurarse de que los Makefiles son correctos. Algunos lo son y algunos no.

Algo importante que hay que advertir es que hay que añadir el servidor proxy al `/etc/inetd.conf`. Se debe añadir la linea:

```
socks   stream  tcp     nowait  nobody  /usr/local/etc/sockd    sockd
```

para decir a `inetd` que arranque el servidor cuando se le pida.

5.1 Configuración del Servidor Proxy

El programa `socks` necesita dos ficheros de configuración distintos. Uno en el que se le dice qué accesos están permitidos, y otro para dirigir las peticiones al servidor proxy apropiado. El fichero de control de acceso debe residir en el servidor. El fichero de rutado debe residir en todas las máquinas Ún*x. Las máquinas DOS y, presumiblemente, las Macintosh encaminarán por sí mismas.

5.2 El Fichero de Control de Acceso

Con socks4.2 Beta, el fichero de acceso se llama `"sockd.conf"`. Debería contener dos tipos de líneas: las de permiso, que contienen "permit" y las de prohibición, que contienen "deny". Cada linea tendrá tres palabras:

- El identificador (permit/deny)
- La dirección IP
- El modificador de dirección

El identificador es o permit (permitir) o deny (denegar). Debería haber uno de cada.

La dirección IP se compone de cuatro octetos según la típica notación de puntos: p.ej. 192.168.2.0 .

El modificador de dirección es también un número de cuatro octetos. Funciona como una máscara de red. Hay que verlo como 32 bits (unos o ceros). Si el bit es uno, el bit correspondiente de la dirección que se comprueba debe coincidir con el bit correspondiente del campo de dirección IP.

Por ejemplo, si la línea es:

```
permit 192.168.2.23 255.255.255.255
```

entonces, admitirá sólo direcciones IP en las que coincidan todos los bits de 193.168.2.23, esto es, sólo ella misma. La línea:

```
permit 192.168.2.0 255.255.255.0
```

admitirá todas las direcciones desde la 192.168.2.0 hasta la 192.168.2.255, la subred de clase C completa. No se debería tener la línea:

```
permit 192.168.2.0 0.0.0.0
```

dado que permitiría cualquier dirección.

Así que, primero, permitimos todas las direcciones que queramos permitir, y después prohibimos el resto. Para permitir a cualquiera del rango 192.168.2.xxx, las líneas:

```
permit 192.168.2.0 255.255.255.0
deny 0.0.0.0 0.0.0.0
```

funcionarán perfectamente. Observa los primeros `"0.0.0.0"` en la linea de prohibición. Con un modificador de 0.0.0.0, el campo de la dirección IP no importa. Se suele poner todo ceros porque es fácil de teclear.

Se puede poner más de una línea de cada clase.

También se puede autorizar o denegar el acceso a determinados usuarios. Se consigue gracias a la autentificación del protocolo *ident*. No todos los sistemas soportan *ident* (incluyendo al Trumpet Winsock) de modo que no profundizaré en ello. La documentación que viene con socks trata este tema adecuadamente.

5.3 El Fichero de rutado

El fichero de rutado de socks tiene el desafortunado nombre de **"socks.conf"**. Y digo que es desafortunado porque se parece tanto al del fichero de control de acceso que es fácil confundirlos.

El fichero de rutado tiene la función de decir a los clientes de socks cuándo usar socks y cuándo no. Por ejemplo, en nuestra red la máquina 192.168.2.3 no necesita usar socks para comunicarse con la 192.168.2.1 (el cortafuegos), ya que tiene una conexión directa vía Ethernet. La 127.0.0.1, dirección de "vuelta atrás" (que representa a una máquina ante ella misma), está definida automáticamente. Está claro que no se necesita usar socks para hablar con uno mismo.

Hay tres tipos de entradas:

- deny
- direct
- sockd

Deny (denegar) le dice a socks que peticiones debe rechazar. Esta entrada tiene los mismos tres campos que en `sockd.conf`, identificador, dirección, y modificador. Generalmente, dado que esto también es manejado por el fichero de control de acceso `sockd.conf`, el modificador se pone a 0.0.0.0 . Si uno quiere impedirse a si mismo conectar con un determinado sitio, se puede hacer poniéndolo aquí.

La entrada `direct` dice para qué direcciones *no* se debe usar socks. Éstas son todas las direcciones a las que se puede llegar sin usar el servidor proxy. De nuevo hay tres campos: identificador, dirección, y modificador. Nuestro ejemplo tendría:

```
direct 192.168.2.0 255.255.255.0
```

Con lo que iría directamente a cualquier máquina de la red protegida.

La entrada `sockd` dice cuál es la máquina en la que corre el servidor de socks. La sintaxis es:

```
sockd @=<lista de servidores> <direccion IP> <modificador>
```

Observemos la entrada `@=`. Ésta permite poner las direcciones IP de una lista de servidores proxy. En nuestro ejemplo sólo usamos un servidor proxy, pero se puede tener muchos para admitir una carga mayor y conseguir redundancia frente a fallos.

La dirección IP y el modificador funcionan como en los otros ejemplos. Especifican a qué direcciones se va a través de los servidores.

5.4 El Servicio de Nombres tras el Cortafuegos

Instalar un Servicio de Nombres detrás de un cortafuegos es relativamente simple. No hay más que instalar el servidor de DNS en la máquina que hace de cortafuegos. Luego se hace que todas las máquinas tras el cortafuegos usen este servidor de DNS.

5.5 Trabajar con un Servidor Proxy

5.5.1 Unix Para que las aplicaciones funcionen con el servidor proxy, hay que ”sockificarlas”. Será necesario tener dos telnets distintos, uno para la comunicación directa, y uno para la comunicación a través del servidor proxy. Socks viene con instrucciones de cómo sockificar un programa, así como con un par de programas ya sockificados. Si se usa la versíon sockificada para conectar con algún sitio al que se tiene acceso directo, socks cambiará automáticamente a la versión para acceso directo (la normal). Por esta razón deberemos cambiar el nombre a todos los programas de la red protegida y sustituirlos por los sockificados. Así ”finger” pasará a ser ”finger.orig”, ”telnet” a ”telnet.orig”, etc... . Se debe dar a conocer a socks todo esto en el fichero include/socks.h . Algunos programas gestionan el rutado y el sockificado ellos mismos. Éste es el caso de *Netscape*. Se puede usar un servidor proxy con *Netscape* simplemente poniendo la dirección del servidor (192.168.2.1 en nuestro caso) en el campo SOCKs del menu Proxys. Todas las aplicaciones necesitarán algún retoque independientemente de cómo manejen la existencia de servidores proxy.

5.5.2 MS Güindous con el Trumpet Winsock

El Trumpet Winsock lleva incorporada la gestión de servidores proxy. En el menú ”setup” se debe poner la dirección IP del servidor y las direcciones de todos los ordenadores a los que se llega directamente. Él se encargará a partir de entonces de todos los paquetes de salida.

5.6 Cómo conseguir que el Servidor Proxy funcione con UDP

El paquete socks sólo funciona con *TCP*, no con *UDP*. Esto le quita un poco de utilidad. Muchos programas interesantes, (como talk o archie) usan UDP. Existe un paquete diseñado para funcionar como un servidor proxy para paquetes de UDP que se llama UDPrelay. El autor es Tom Fitzgerald fitz@wang.com. Desgraciadamente, en el momento de escribir estas líneas, no es compatible con LiNUX.

5.7 Inconvenientes de los Servidores Proxy

Un servidor proxy es ante todo un *dispositivo de seguridad*. Usarlo para aumentar el número de máquinas con acceso a la Internet cuando se tienen pocas direcciones IP tiene muchos inconvenientes. Un servidor proxy permite un mayor acceso desde dentro de la red protegida al exterior, pero mantiene el interior completamente inaccesible desde el exterior. Esto significa que no habrá conexiones archie, ni talk, ni envío directo de correo a los ordenadores de dentro. Estos inconvenientes pueden parecer pequeños, pero consideremos los siguientes casos:

- Te has dejado un informe que estás haciendo en tu ordenador que está dentro de la red protegida por el cortafuegos. Estás en casa y decides cambiar algo. No puedes. No puedes llegar a tu ordenador dado que está tras el cortafuegos. Intentas entrar en el `cortafuegos` primero, pero como todo el mundo tiene acceso al exterior gracias al servidor proxy, no te han abierto cuenta en él.
- Tu hija va a la universidad. Quieres enviarle correo. Tienes algunas cosas privadas que comentar con ella, por lo que preferirías que el correo llegara directamente a tu máquina. Confías plenamente en el administrador de tu sistema, pero, aún así, es correo privado.
- La incapacidad de manejar paquetes UDP es un gran inconveniente de los servidores proxy. Imagino que no por mucho tiempo.

El FTP crea otro problema con los servidores proxy. Cuando se hace un `ls`, o un `get`, el servidor de FTP establece una conexión con la máquina cliente y manda la información por ella. Un servidor proxy no lo permitirá, así que el FTP no funciona especialmente bien.

Además, un servidor proxy es lento. Debido a la gran sobrecarga, casi cualquier otro medio de lograr acceso será más rápido.

Resumiendo, si tienes suficientes direcciones IP y no te preocupa la seguridad, no uses cortafuegos ni servidores proxy. Si no tienes suficientes direcciones IP, pero tampoco te preocupa la seguridad, seguramente deberías considerar los "emuladores de IP" como Term, Slirp, o TIA.

Term se puede conseguir en `ftp://sunsite.unc.edu`, Slirp en `ftp://blitzen.canberra.edu.au/pub/slirp`, y TIA en `ftp://marketplace.com`.

Van más rápido, permiten mejores conexiones, y proveen un mayor nivel de acceso a la red interior desde la Internet. Los servidores proxy están bien para las redes que tienen muchos ordenadores que quieren conectar con la Internet al vuelo, y en las que se quiere poco trabajo de mantenimiento tras la instalación.

6 Configuración Avanzada

Hay una configuración que me gustaría mostrar antes de dar por terminado este documento. La que acabo de comentar bastará seguramente para la mayoría de la gente. Sin embargo, pienso que el próximo ejemplo mostrará una más avanzada que puede resolver algunas dudas. Si tienes preguntas que trascienden lo cubierto hasta aquí, o simplemente estás interesado/a en la versatilidad de los servidores proxy y los cortafuegos, sigue leyendo.

6.1 Una gran red con énfasis en la seguridad

Digamos, por ejemplo, que eres el líder de la *Vigésimo Tercera Hermandad de la Discordia de Milwaukee*. Te gustaría poner una red. Tienes 50 ordenadores y una subred de 32 (5 bites) direcciones IP (reales). Hay varios niveles de acceso. Se dicen cosas distintas a los discípulos según el nivel en que están. Obviamente, querrás proteger ciertas partes de la red de los discípulos que no están en ese nivel.

Renuncia de Responsabilidad: No soy miembro de la Hermandad de la Discordia. No conozco su terminología, ni me importa. Solo los estoy usando como ejemplo. Por favor, mandad todos los frutos de vuestros arrebatos de ira a

Los niveles son:

1. **El nivel externo**. Éste es el nivel que se enseña a cualquiera. Básicamente es un rollo patatero sobre Eris, Diosa de la Discordia, y un montón de chorradas más.
2. **Iniciado**. Este es el nivel para la gente que ha pasado del nivel externo. Aquí es donde se les dice que la discordia y la estructura son realmente una, y que Eris es también Jehová.
3. **Adepto**. Aquí es donde se encuentra el *verdadero* plan. En este nivel se guarda toda la información de cómo la Sociedad de la Discordia va a dominar el mundo gracias a un diabólico, aunque jocoso, plan que implica a Newt Gingrich, los Cereales Wheaties, O.J. Simpson, y quinientos cristales de cuarzo erróneamente etiquetados como de 6,5 MHz.

6.1.1 Configuración de la Red

Las direcciones IP se disponen así:

- Una dirección es 192.168.2.255, que es la de difusión y por lo tanto no utilizable.
- 23 de las 32 direcciones IP se asignan a las 23 máquinas accesibles desde la Internet.
- Una dirección IP extra es para una máquina LiNUX en esa red.
- Una dirección IP extra es para otra máquina LiNUX en esa red.
- Dos direcciones IP son para el router que los conecta con la Internet.
- Cuatro se dejan sin usar, pero se les asignan los nombres paul, ringo, john, y george, sólo para confundir las cosas un poco.
- Las dos redes protegidas tienen direcciones del tipo 192.168.2.xxx .

Entonces se instalan dos redes, cada una en una habitación separada. Se utilizan Ethernets de infrarrojos, de tal manera que son completamente invisibles desde la habitación exterior. Por suerte, la Ethernet de infrarrojos funciona como la normal (o eso creo), de modo que podemos pensar en ellas como si fueran Ethernets normales.

Cada una de esas redes se conecta a una de las máquinas LiNUX a las que se asignaron las direcciones IP extras.

Hay un servidor de ficheros que conecta las dos redes protegidas. Esto se debe a que los planes para dominar el mundo implican a algunos de los iniciados de mayor nivel. El servidor de ficheros tiene la dirección `192.168.2.17` para la red de iniciados, y la `192.168.2.23` para la de adeptos. Tiene que tener dos direcciones dado que tiene dos tarjetas Ethernet. Tiene **deshabilitado** el reenvio de paquetes IP.

El reenvio de paquetes IP también está deshabilitado en los dos LiNUXes. El router no encaminará paquetes con destino 192.168.2.xxx a menos que se le diga explícitamente, así que la Internet en ningún caso podría acceder al interior. La razón para deshabilitar el reenvio de paquetes IP aquí es para que los paquetes de la red de adeptos no lleguen a la de iniciados y viceversa.

El servidor de NFS puede ser configurado para ofrecer diferentes ficheros a las diferentes redes. Esto puede venir al pelo, y unos pocos trucos con enlaces simbólicos pueden hacer que se compartan los ficheros comunes entre todos. Con esta configuración y otra tarjeta Ethernet, el mismo servidor de ficheros puede dar servicio a las tres redes.

6.1.2 El Servidor Proxy

Dado que los tres niveles quieren rastrear la Internet para sus propios y diabólicos propósitos, los tres necesitan tener acceso a ella. La red externa está conectada directamente a la Internet, luego no tenemos que hacer nada. Las redes de adeptos e iniciados están detrás de sendos cortafuegos, luego es necesario instalar servidores proxy para ellas.

Las dos redes se configurarán de forma muy parecida. Ambas tienen las mismas direcciones IP asignadas. Añadiré un par de requisitos para hacerlo más interesante:

1. No se debe poder usar el servidor de ficheros para acceder a la Internet Esto le expone a virus y otras cosas desagradables, y es bastante importante.
2. No permitiremos a los Iniciados acceso al World Wide Web. Están formándose, y la adquisición de ese tipo de información podría resultar dañina.

Así, el fichero `sockd.conf` en el LiNUX de los iniciados tendrá esta línea:

```
deny 192.168.2.17 255.255.255.255
```

y en la máquina de los adeptos:

```
deny 192.168.2.23 255.255.255.255
```

Y, el LiNUX de los iniciados tendrá esta línea

```
deny 0.0.0.0 0.0.0.0 eq 80
```

Que dice que se deniegue a todas las máquinas el acceso al puerto igual (*eq*) a 80, el puerto del http. Esto aún permitirá el acceso a otros servicios, sólo impedirá el acceso al Web.

Además, ambos ficheros contendrán:

```
permit 192.168.2.0 255.255.255.0
```

para permitir a todos los ordenadores de la red 192.168.2.xxx usar este servidor proxy, excepto aquello que ya ha sido prohibido (esto es: cualquier acceso desde el servidor de ficheros y el acceso al Web desde la red de iniciados)

El fichero `sockd.conf` de los iniciados será más o menos:

```
deny 192.168.2.17 255.255.255.255
deny 0.0.0.0 0.0.0.0 eq 80
permit 192.168.2.0 255.255.255.0
```

y el de los adeptos será más o menos:

```
deny 192.168.2.23 255.255.255.255
permit 192.168.2.0 255.255.255.0
```

Con esto todo debería estar configurado correctamente. Cada red está aislada como corresponde, con el grado apropiado de interacción. Todo el mundo debería estar contento. Ahora, *cuidado con los cristales de 6,5 MHz...*

7 Anexo: El INSFLUG

El *INSFLUG* forma parte del grupo internacional *Linux Documentation Project*, encargándose de las traducciones al castellano de los Howtos (Comos), así como la producción de documentos originales en aquellos casos en los que no existe análogo en inglés.

En el **INSFLUG** se orienta preferentemente a la traducción de documentos breves, como los *COMOs* y *PUFs* (**P**reguntas de **U**so **F**recuente, las *FAQs*. `:)`), etc.

Diríjase a la sede del INSFLUG para más información al respecto.

En la sede del INSFLUG encontrará siempre las **últimas** versiones de las traducciones: `www.insflug.org`. Asegúrese de comprobar cuál es la última versión disponible en el Insflug antes de bajar un documento de un servidor réplica.

Se proporciona también una lista de los servidores réplica (*mirror*) del Insflug más cercanos a Vd., e información relativa a otros recursos en castellano.

Francisco José Montilla, `pacopepe@insflug.org`.