FAQ sobre Linux para principiantes - es.comp.os.linux

Rafael Martinez, r.m.guerrero@medisin.uio.no
A. Gustavo Gonzalez, agonzale@cica.es
Mario Teijeiro Otero, asimovi@teleline.es
Manuel Constantino, constan@bart.us.es
Santiago Romero Iglesias, santiago.romero@iname.com
02 Julio 1999 / Version 1.0

Este documento intentara dar respuesta a las preguntas y problemas mas comunes que un principiante se hace antes de instalar linux, y al empezar a trabajar con Linux.

Índice General

1 Introduccion e informacion general

1.1 Introduccion

Este documento pretende dar respuesta, a las preguntas mas habituales, que los usuarios suelen hacerse cuando empiezan a trabajar con Linux. La idea surgio en el grupo de noticias es.comp.os.linux. La FAQ ha sido creada por el coordinador de la misma, *Rafael Martinez* `<mailto:r.m.guerrero@medisin.uio.no>` y por los colaboradores de la FAQ, *A. Gustavo Gonzalez* `<mailto:agonzale@cica.es>`, *Mario Teijeiro Otero* `<mailto:asimovi@teleline.es>`, *Manuel Constantino* `<mailto:constan@bart.us.es>`, *Santiago Romero Iglesias* `<mailto:santiago.romero@iname.com>`.

La pagina principal de la FAQ es *http://linux-es.uio.no/Faq/* `<http://linux-es.uio.no/Faq/>`, aqui podreis encontrar la ultima version de la misma en formato HTML, PS, DVI, TEXTO y SGML. Tambien es posible encontrarla en *LUCAS* `<http:/lucas.hispalinux.es>` y en los servidores espejos del mismo.

Si quereis colaborar y aportar ideas o sugerencias, ponerse en contacto con el *coordinador* `<mailto:r.m.guerrero@medisin.uio.no>` de este documento.

Debido a la extension de esta FAQ y a las ideas para ampliarla, puede que para la proxima version se cambie el nombre a «Guía de resolución de problemas, (GREP)» incorporandose a la seccion de guias y manuales de Lucas.

Esperamos que este documento os ayude a encontrar respuesta a vuestro problema y que vuestro sistema mejore dia a dia.

1.2 ¿Qué es Linux?

LINUX es un sistema operativo, compatible Unix. Dos caracteristicas muy peculiares lo diferencian del resto de los sistemas que podemos encontrar en el mercado, la primera, es que es *libre*, esto significa que no tenemos que pagar ningun tipo de licencia a ninguna casa desarrolladora de software por el uso del mismo, la segunda, es que el sistema viene acompañado del *codigo fuente*. El sistema lo forman el Nucleo (Leer seccion 5 (Kernel/Nucleo)) del sistema mas un gran numero de programas / librerias que hacen posible su utilizacion.

Linux se distribuye bajo la *GNU General Public License* `<http://www.gnu.org/copyleft/gpl.html>`, por lo tanto, el codigo fuente tiene que estar siempre accesible.

El sistema ha sido diseñado y programado por multitud de programadores alrededor del mundo. El nucleo del sistema sigue en continuo desarrollo bajo la coordinacion de Linus Torvalds, la persona de la que partio la idea de este proyecto, a principios de la decada de los noventa.

Dia a dia, mas y mas programas / aplicaciones estan disponibles para este sistema, y la calidad de los mismos aumenta de version a version. La gran mayoria de los mismos vienen acompanados del codigo fuente y se distribuyen gratuitamente bajo los terminos de licencia de la GNU Public License.

En los ultimos tiempos, ciertas casas de software comercial han empezado a distribuir sus productos para Linux y la presencia del mismo en empresas aumenta rapidamente por la excelente relacion calidad-precio que se consigue con Linux.

Las plataformas en las que en un principio se puede utilizar Linux son 386-, 486-. Pentium, Pentium Pro, Pentium II,Amiga y Atari, tambien existen versiones para su utilizacion en otras plataformas, como Alpha, ARM, MIPS, PowerPC y SPARC

1.3 Historia de Linux

LINUX hace su aparicion a principios de la decada de los noventa, era el año 1991 y por aquel entonces un estudiante de informatica de la Universidad de Helsinki, llamado Linus Torvalds empezo, -como una aficion y sin poderse imaginar a lo que llegaria este proyecto, a programar las primeras lineas de codigo de este sistema operativo llamado LINUX.

Este comienzo estuvo inspirado en MINIX, un pequeño sistema Unix desarrollado por Andy Tanenbaum. Las primeras discusiones sobre Linux fueron en el grupo de noticias comp.os.minix, en estas discusiones se hablaba sobre todo del desarrollo de un pequeño sistema Unix para usuarios de Minix que querian mas.

Linus nunca anuncio la version 0.01 de Linux (agosto 1991), esta version no era ni siquiera ejecutable, solamente incluia los principios del nucleo del sistema, estaba escrita en lenguaje ensamblador y asumia que uno tenia acceso a un sistema Minix para su compilacion.

El 5 de octubre de 1991, Linus anuncio la primera version "Oficial"de Linux, -version 0.02. Con esta version Linus pudo ejecutar Bash (GNU Bourne Again Shell) y gcc (El compilador GNU de C) pero no mucho mas funcionaba. En este estado de desarrollo ni se pensaba en los terminos soporte, documentacion, distribucion .....

Despues de la version 0.03, Linus salto en la numeracion hasta la 0.10, mas y mas programadores a lo largo y ancho de internet empezaron a trabajar en el proyecto y despues de sucesivas revisiones, Linus incremento el numero de version hasta la 0.95 (Marzo 1992). Mas de un año despues (diciembre 1993) el nucleo del sistema estaba en la version 0.99 y la version 1.0 no llego hasta el 14 de marzo de 1994.

Desde entonces no se ha parado de desarrollar, la version actual del nucleo es la 2.2 y sigue avanzando dia a dia con la meta de perfeccionar y mejorar el sistema.

1.4 Caracteristicas de Linux

Aqui teneis una lista bastante completa con las caracteristicas de LINUX
[Fuente: Infosheet-Como. Autor: Ivan Casado] :

- Multitarea: La palabra multitarea describe la habilidad de ejecutar varios programas al mismo tiempo. LINUX utiliza la llamada multitarea preeventiva, la cual asegura que todos los programas que se estan utilizando en un momento dado seran ejecutados, siendo el sistema operativo el encargado de ceder tiempo de microprocesador a cada programa.
- Multiusuario: Muchos usuarios usando la misma maquina al mismo tiempo.
- Multiplataforma: Las plataformas en las que en un principio se puede utilizar Linux son 386-, 486-. Pentium, Pentium Pro, Pentium II,Amiga y Atari, tambien existen versiones para su utilizacion en otras plataformas, como Alpha, ARM,MIPS, PowerPC y SPARC.
- Multiprocesador: Soporte para sistemas con mas de un procesador esta disponible para Intel y SPARC.
- Funciona en modo protegido 386.
- Protección de la memoria entre procesos, de manera que uno de ellos no pueda colgar el sistema.
- Carga de ejecutables por demanda: Linux sólo lee del disco aquellas partes de un programa que están siendo usadas actualmente.

- Política de copia en escritura para la compartición de páginas entre ejecutables: esto significa que varios procesos pueden usar la misma zona de memoria para ejecutarse. Cuando alguno intenta escribir en esa memoria, la página (4Kb de memoria) se copia a otro lugar. Esta política de copia en escritura tiene dos beneficios: aumenta la velocidad y reduce el uso de memoria.
- Memoria virtual usando paginación (sin intercambio de procesos completos) a disco: A una partición o un archivo en el sistema de archivos, o ambos, con la posibilidad de añadir más áreas de intercambio sobre la marcha Un total de 16 zonas de intercambio de 128Mb de tamaño máximo pueden ser usadas en un momento dado con un límite teórico de 2Gb para intercambio. Este limite se puede aumentar facilmente con el cambio de unas cuantas lineas en el codigo fuente.
- La memoria se gestiona como un recurso unificado para los programas de usuario y para el caché de disco, de tal forma que toda la memoria libre puede ser usada para caché y ésta puede a su vez ser reducida cuando se ejecuten grandes programas.
- Librerías compartidas de carga dinámica (DLL's) y librerías estáticas.
- Se realizan volcados de estado (core dumps) para posibilitar los análisis post-mortem, permitiendo el uso de depuradores sobre los programas no sólo en ejecución sino también tras abortar éstos por cualquier motivo.
- Compatible con POSIX, System V y BSD a nivel fuente.
- Emulación de iBCS2, casi completamente compatible con SCO, SVR3 y SVR4 a nivel binario.
- Todo el código fuente está disponible, incluyendo el núcleo completo y todos los drivers, las herramientas de desarrollo y todos los programas de usuario; además todo ello se puede distribuir libremente. Hay algunos programas comerciales que están siendo ofrecidos para Linux actualmente sin código fuente, pero todo lo que ha sido gratuito sigue siendo gratuito.
- Control de tareas POSIX.
- Pseudo-terminales (pty's).
- Emulación de 387 en el núcleo, de tal forma que los programas no tengan que hacer su propia emulación matemática. Cualquier máquina que ejecute Linux parecerá dotada de coprocesador matemático. Por supuesto, si el ordenador ya tiene una FPU (unidad de coma flotante), esta será usada en lugar de la emulación, pudiendo incluso compilar tu propio kernel sin la emulación matemática y conseguir un pequeño ahorro de memoria.
- Soporte para muchos teclados nacionales o adaptados y es bastante fácil añadir nuevos dinámicamente.
- Consolas virtuales múltiples: varias sesiones de login a través de la consola entre las que se puede cambiar con las combinaciones adecuadas de teclas (totalmente independiente del hardware de video). Se crean dinámicamente y puedes tener hasta 64.
- Soporte para varios sistemas de archivo comunes, incluyendo minix-1, Xenix y todos los sistemas de archivo típicos de System V, y tiene un avanzado sistema de archivos propio con una capacidad de hasta 4 Tb y nombres de archivos de hasta 255 caracteres de longitud.
- Acceso transparente a particiones MS-DOS (o a particiones OS/2 FAT) mediante un sistema de archivos especial: no es necesario ningún comando especial para usar la partición MS-DOS, esta parece un sistema de archivos normal de Unix (excepto por algunas restricciones en los nombres de archivo, permisos, y esas cosas). Las particiones comprimidas de MS-DOS 6 no son accesibles en este momento, y no se espera que lo sean en el futuro. El soporte para VFAT (WNT, Windows 95) ha sido añadido al núcleo de desarrollo y estará en la próxima versión estable.

- Un sistema de archivos especial llamado UMSDOS que permite que Linux sea instalado en un sistema de archivos DOS.
- Soporte en sólo lectura de HPFS-2 del OS/2 2.1
- Sistema de archivos de CD-ROM que lee todos los formatos estándar de CD-ROM.
- TCP/IP, incluyendo ftp, telnet, NFS, etc.
- Appletalk.
- Software cliente y servidor Netware.
- Lan Manager / Windows Native (SMB), software cliente y servidor.
- Diversos protocolos de red incluidos en el kernel: TCP, IPv4, IPv6, AX.25, X.25, IPX, DDP, Netrom, etc.

1.5 Linux Copyright

El copyright del kernel de linux pertenece a Linus Torvalds. Este se distribuye bajo la GNU General Public License `<http://www.gnu.org/copyleft/gpl.html>`, esto significa basicamente, que cualquiera puede libremente, copiarlo, cambiarlo y distribuirlo, pero sin posibilidad de aplicar restricciones en futuras distribuciones y teniendo que incluir el codigo fuente.

Las licencias de los programas y aplicaciones que acompañan al nucleo varian, muchos pertenecen al proyecto GNU y se distribuyen bajo GPL.

2 Antes de la instalacion

2.1 ¿Es linux para mi?

Aunque parezca muy simple, esta pregunta conlleva otra serie de cuestiones anejas, que hemos de considerar para responder con verosimilitud al profano en este sistema. Por otra parte, aunque convencidos abogados de este sistema operativo frente a los que no son de código abierto, también hemos de aceptar las limitaciones del mismo con respecto a los nuevos usuarios.

En un principio, linux era para gurús, hackers y desarrolladores...y hay muchos que piensan que debería seguir siendo así. No obstante, las facilidades que tanto en instalación, configuración y gestión del sistema, han proporcionado las distribuciones en sus versiones más recientes han acercado mucho al usuario final al mundo linux.

Cuando pensamos en esta FAQ, lo hicimos considerando que los "novatos"en este sistema podían ser de diversas procedencias; algunos podrían ser grandes conocedores de informática, programadores y administradores de sistemas "no-linux", quienes se adaptarían fácilmente. Otros podrían ser usuarios finales de algún otro sistema operativo, que utilizan el ordenador para correr aplicaciones, como algunos conductores usan el vehículo sólo como medio para acceder a lugares lejanos en tiempos cortos, pero no quieren saber nada de mecánica ni electricidad del automóvil...Al preguntarnos ¿Es linux para mí? queremos decir más bien *¿Me conviene usar Linux? ¿Para que lo quiero?*

Hemos leído muchos libros y un gran número de documentos acerca de linux en relación con el usuario. Uno de ellos, titulado @"de Todd Burgess (***http://eddie.cis.uoguelph.ca/~tburgess*** `<http://eddie.cis.uoguelph.ca/~tburgess>`) trata en profundidad el tema de los tránsfugas de sistemas operativos por cuestiones de moda (Cool people use linux=la gente güai usa linux) y nos dice:

Usa linux:

- Si lo requiere la aplicación que necesitas
- Si deseas aumentar tu conocimiento acerca de sistemas operativos
- Si no te importa investigar "per se", como imperativo categórico

No uses linux:

- Si no estás ducho en sistemas operativos
- Solo porque no es Microsoft
- Si piensas que por eso eres más güai, inteligente o mejor que el usuario medio

En una línea similar, encontramos otro interesante documento que (*http://source.syr.edu/~jdimpson/linux/usage/nouse.html* <http://source.syr.edu/~jdimpson/linux/usage/nouse.html>) mantenía que:

No deberíamos usar linux:

- Si nuestras necesidades están cubiertas con nuestro sistema actual y pensamos que continuará así en el futuro.
- Si trabajamos en un ambiente comercial/industrial/profesional donde no se usa linux y no disponemos de apoyo técnico por parte de nuestra plantilla técnica ni pensamos poder proporcionar dicho apoyo nosotros mismos.
- Si alguna aplicación importante y necesaria para nuestro trabajo no está disponible en linux ni hay sustitutos adecuados.

Mucha gente que lea esta FAQ en el futuro, provendrá del ambiente Microsoft (Windows 95/98/NT/00 ;-)). En tal caso, aquéllos que usaren su computador básicamente para el procesamiento de textos y el empleo de hojas de cálculo, deberían quedarse como están. A pesar de todos sus fallos, Windows es fácil de usar, más aún de aprender y *Dispone de una ingente cantidad de softtware*. Los que tienen otras ambiciones, pueden *plantearse cambiar a linux por 10 importantes razones* (*http://www.croftj.net/~goob/local/why.html* <http://www.croftj.net/~goob/local/why.html>)

1. Las distribuciones de Linux no son costosas: pueden obtenerse gratis de los lugares apropiados de internet o bien comprar los CDs por muy poco dinero comparado con otros sistemas comerciales.
2. Linux es un sistema operativo completo con tres grandes características: *Estable* (es muy raro que colapse una aplicación en linux), *Fidedigno* (un servidor linux puede funcionar durante cientos de días sin necesidad de reiniciarse, lo que no es común en sistemas Windows) y *Extremadamente potente*.
3. Tiene un entorno de programación completo, incluyendo C, C++, compiladores Fortran, utilidades como Qt y lenguajes de guiones (scripts) como Perl, gwak y sed. (un simple compilador C para Windows cuesta cientos de dólares)
4. Presenta una excelentes prestaciones para conexión en red, permitiendo compartir PPUs y módems.
5. Constituye el entorno ideal para hacer funcionar servidores web (Apache) o FTP (oran).
6. Además del software "libre", hay tambien bastante software comercial para linux.
7. Se actualiza y mantiene fácilmente (a diferencia de Windows).
8. Admite múltiples procesadores como norma.

9. Es realmente multitarea (a diferencia de Windows),
10. Tiene un excelente servidor X-windows y numerosos gestores de ventanas X.

y además, por instalar linux no tendremos que renunciar a Windows: es posible gestionar linux + otros sistemas operativos en el mismo ordenador.

Para terminar, me gustaría copiar literalmente un trozo de la introducción de un libro titulado @"de Raúl Montero Rivero (Ed. Anaya) que viene como anillo al dedo al nuevo usuario de nuestro sistema:

"Aprenderás qué es linux, sus virtudes y sus defectos, cómo puedes conseguirlo, qué distribución elegir a la hora de instalarlo, cómo administrar tu sistema correctamente, cómo trabajar con X-windows, cómo conectarte con él a Internet...Pero no todo va a ser tan fácil con linux, porque como comprobarás, no es un sistema hecho para cobardes. Tendrás que ser valiente"

Si te da miedo formatear tu disco duro, linux no es para tí. Si te acobarda tener que reinstalar todo de nuevo sólo porque has cometido un pequeño error en el proceso de instalación, abandona y no sigas. Si te aterra la mera idea de particionar tu disco duro, apaga y vámonos. Y si te vas pasar todo el rato pensando que al configurar la frecuencia de tu monitor para que funcione X windows, te lo puedes cargar, mejor que vuelvas a tu Windows de siempre.

Ahora seguro que ya sabes si linux es para tí... ¡Bienvenido!

2.2 ¿Cuanto espacio necesito en mi disco duro, para instalar Linux?

Esto depende en gran medida de la cantidad de programas/paquetes que quieras instalar, del espacio que reserves para swap (espacio de intercambio) y del espacio libre que quieras tener para datos/documentos. Hay que puntualizar que los datos que se dan en esta subseccion hacen referencia al sistema en si, mas programas.

- Una instalacion super minima, necesitara alrededor de unos 10MB.
- Una instalacion minima con las X incluidas, alrededor de los 80Mb.
- Para tener un sistema completo (sistema/librerias/programas/aplicaciones/espacio para almacenamiento de datos), funcional y siendo realistas, habria que tener de 500Mb a 1,5Gb de espacio en el disco duro. Como en cualquier sistema, si quieres instalar todos los programas que pasen por tus manos o si la cantidad de informacion con la que trabajas es muy grande, necesitaras mas y mas espacio y siempre te parecera poco el que tienes ;-)

2.3 ¿Cuanta memoria necesito, para usar Linux?

Esto dependera tambien en gran medida del uso que le vayas a dar a tu maquina, cuanta mas memoria, mejor trabajara. Aqui tienes unos datos orientativos sobre la cantidad de memoria necesaria:

- Como minimo se necesitan 4Mb de memoria RAM para utilizar Linux, esta cantidad es como se indica minima y no apta para trabajar con sistemas graficos.
- Para trabajar con las X-windows de una manera decente (dependera del entorno grafico utilizado) se necesitara como minimo 16Mb.
- Para un uso intensivo y para trabajar con programas que requieren muchos recursos, se recomienda de 32Mb en adelante. Cuanta mas memoria se tenga, mas suelto trabajara nuestro sistema y mas programas podremos tener cargados en memoria.

2.4 ¿Funcionará mi equipo con linux?

Hoy en dia, la gran mayoria de hardware esta soportado por Linux, asi que no deberiais tener problemas. De todas maneras, existen configuraciones que no estan soportadas y quizas ultimos modelos de tarjetas graficas, controladoras, etc, que tampoco lo esten.

Dia a dia se trabaja para dar soporte a nuevo hardware y lo que no este soportado hoy, probablemente este soportado muy pronto. En un pasado, muchas casas de productos no apoyaban a linux, con el consiguiente retraso en el soporte a estos productos, con respecto a otros sistemas operativos, pero ultimamente esto esta cambiando rapidamente.

Si quereis comprobar si vuestro hardware esta soportado por linux o no, leer el *Hardware-Howto* `<http://users.bart.nl/~patrickr/hardware-howto/Hardware-HOWTO.html>`, un documento con el hardware soportado por linux. Tambien os podeis pasar por esta excelente web, *Linuxhardware.net* `<http://www.linuxhardware.net/>`, donde podreis consultar sobre vuestro hardware.

2.5 ¿Cómo consigo Linux?

Linux es una sistema de libre distribucion por lo que podeis encontrar todos los ficheros necesarios para su funcionamiento en multitud de servidores conectados a Internet. Casi todos los principales distribuidores de Linux, ofrecen la posibilidad de bajarse sus distribuciones, via FTP.

Otra alternativa para los que van a instalar el sistema por primera vez son las llamadas, "Distribuciones"de Linux. Una distribucion no es otra cosa que una recopilacion de programas y ficheros, organizados y preparados para su instalacion. Estas distribuciones se pueden obtener a traves de Internet, como ya hemos explicado, pero tambien se encuentran en varios CDes, conteniendo todos los ficheros necesarios para instalar un sistema Linux bastante completo y en la mayoria de los casos un programa de instalacion que nos ayudara en la tarea de una primera instalacion.

Existen varias distribuciones creadas por diferentes empresas a unos precios bastantes asequibles, las cuales deberiais de poder encontrar en tiendas de informatica o librerias. En el peor de los casos siempre podeis encargarlas directamente por Internet a las empresas que las crean. A veces, las revistas de informatica sacan una edicion bastante aceptable de alguna distribucion.

Si vais a instalar el sistema por primera vez, os recomiendo haceros con una de estas distribuciones y en un futuro cuando querais actualizar el sistema con las ultimas versiones y actualizaciones del nucleo y programas que utiliceis, usar Internet.

Para mas informacion sobre las distintas distribuciones de Linux podeis leer la *Linux Distribution HOWTO* `<http://metalab.unc.edu/mdw/HOWTO/Distribution-HOWTO.html>`

A continuacion teneis una lista con las principales distribuciones:

- **Redhat**
  Pagina web: *http://www.redhat.com* `<http://www.redhat.com>`
  FTP principal: *ftp://ftp.redhat.com/pub/* `<ftp://ftp.redhat.com/pub>`

- **Debian**
  Pagina web: *http://www.debian.org/* `<http://www.debian.org/>`
  FTP principal: *ftp://ftp.debian.org/debian/* `<ftp://ftp.debian.org/debian/>`

- **SuSE**
  Pagina web: *http://www.suse.de/es/* `<http://www.suse.de/es/>`

FTP principal: *ftp://ftp.suse.com/* <ftp://ftp.suse.com/>

- **Caldera Openlinux**
  Pagina web: *http://www.caldera.com/* <http://www.caldera.com/>
  FTP principal: *ftp://ftp.caldera.com/pub/OpenLinux/* <ftp://ftp.caldera.com/pub/OpenLinux/>

- **Slackware**
  Pagina web: *http://www.slackware.com/* <http://www.slackware.com/>
  FTP principal: *ftp://ftp.cdrom.com/pub/linux/slackware* <ftp://ftp.cdrom.com/pub/linux/slackware>

- **Mklinux**
  Pagina web: *http://www.mklinux.apple.com/* <http://www.mklinux.apple.com/>
  FTP principal: *ftp://ftp.mklinux.apple.com/pub/* <ftp://ftp.mklinux.apple.com/pub/>

- **Eurielec**
  Pagina web: *http://www.eurielec.etsit.upm.es/linux/* <http://www.eurielec.etsit.upm.es/linux/>
  FTP principal: *ftp://ftp.dit.upm.es/linux/* <ftp://ftp.dit.upm.es/linux/>

- **Esware**
  Pagina web: *http://www.esware.com/* <http://www.esware.com/>
  FTP principal: *ftp://ftp.hardlogic.es/pub/* <ftp://ftp.hardlogic.es/pub/>

2.6 ¿Qué distribucion es la mejor?

Esta pregunta se podria denominar *"la pregunta del millon"*. Son muchos los usuarios que la hacen y la respuesta no es facil de hacer.

Como ya se ha explicado en la seccion 1.2 (Que es Linux?), un sistema Linux esta formado por el nucleo (Leer seccion 5 (kernel/nucleo)) mas una serie de programas y librerias que hacen posible tener un sistema productivo y operativo. Tanto el nucleo, como las librerias y programas necesarios estan disponibles libremente en internet.

Con esto, cualquiera con conocimientos, puede instalarse un sistema Linux sin necesidad de adquirir una distribucion, o crear una nueva distribucion desde cero. No hace falta decir, que hacer esto es una tarea que requiere muchos conocimientos y tiempo. Lo que hacen las empresas que crean una distribucion es precisamente esto, poner todos los componentes necesarios juntos, con esto nos evitamos el tener que hacerlo nosotros.

Las distribuciones suelen traer un programa de instalacion, generalmente creado por la empresa que crea la distribucion, manuales con documentacion, una serie de programas/librerias para poder usar el sistema, un kernel, y en algunas ocasiones aplicaciones diversas y soporte tecnico.

Sabido esto, podemos afirmar que la diferencia entre una distribucion u otra es la calidad de lo incluido en la misma; algunos programas de instalacion pueden ser mas intuitivos que otros o mas faciles de utilizar, la calidad de la documentacion, que el conjunto de programas/librerias esten bien testeados y funcionen bien juntos (diferentes versiones, librerias, etc), soporte eficiente, etc, el kernel es el mismo en todas las distribuciones, excepto en la version, es decir, una distribucion A con el kernel 2.2.9 y otra distribucion distinta B con el kernel 2.2.9 estan utilizando exactamente el mismo kernel, el kernel es comun a todas las distribuciones y no existe un kernel para una distribucion y otro para otra.

A continuacion teneis unos comentarios sobre las distribuciones mas usadas. No son en absoluto todas, pero si son las que forman el mayor porcentaje de sistemas Linux instalados:

Redhat:

- Mas extendida, buena calidad.
- Facil de instalar, incluye programas de configuracion que simplifican ciertas tareas.
- Buena documentacion (en ingles).
- Buen seguimiento de fallos y correcciones de los mismos pueden bajarse de su web.

Debian:

- Muy buena calidad. No esta sometida a presiones comerciales, cuidan mucho la calidad antes de su lanzamiento.
- Definida tipicamente como la distribucion de los usuarios avanzados. Puede ser un poco dificil de instalar para el usuario iniciado y sin conocimientos de informatica.
- Buena documentacion (en ingles).
- Buen seguimiento de fallos y correcciones de los mismos pueden bajarse de su web.

SuSE:

- Buena calidad germana.
- Facil de instalar, incluye programas de configuracion que simplifican ciertas tareas.
- Buena documentacion (gran parte en castellano).
- Buen seguimiento de fallos y correcciones de los mismos pueden bajarse de su web.

Caldera:

- Buena calidad.
- Facil de instalar.
- Buena documentacion (en ingles).
- Buen seguimiento de fallos y correcciones de los mismos pueden bajarse de su web

Slackware:

- Esta distribucion fue de las primeras y durante un largo tiempo fue la que mas se utilizo. Ha tenido un tiempo en que no se ha actualizado mucho y eso le ha hecho perder seguidores. Parece que ha vuelto con fuerza.
- Un poco mas complicada de configurar que la media.
- Documentacion en ingles.

2.7 ¿Puedo tener mas de un sistema operativo en mi equipo?

La respuesta es si, no es ningun problema el tener mas de un sistema operativo en tu maquina. Es mas hay mucha gente que tiene incluso 3 y 4 sistemas.

Para hacer esto, hay que tener claro como se instalan los sistemas que queremos tener, como realizar particiones en el disco duro (Leer seccion 3 (Durante la instalacion)) y en que orden deberiamos instalarlos para que no tengamos problemas.

Existen unos documentos Howtos y MiniHowtos que explican muy bien como tener diferentes sistemas en tu ordenador, los podreis encontrar en cualquier servidor que contenga la documentacion de Linux. Mas informacion en la subseccion 14.2 (¿Dónde encuentro documentacion sobre linux?):

- Linux-DOS-Win95-OS2
- Linux + FreeBSD
- NT Loader+Linux
- Linux+Win95
- Loadlin+Win95

2.8 ¿Funcionan mis programas/juegos para Windows en Linux?

Como primera afirmacion, podemos decir que no, tus programas para Windows no funcionaran en Linux. Windows y Linux/Unix no son compatibles y programas compilados en una u otra plataforma no funcionaran en otra plataforma que no sea en la que el programa se compilo para su utilizacion.

Como segunda afirmacion, podemos decir que en algunos casos es posible ejecutar programas para Windows en Linux, si hacemos uso de un emulador de windows para Linux. Un emulador es un programa que se ejecuta en Linux y que crea una maquina virtual windows, engañando al programa windows, que creera que se esta ejecutando en un sistema Windows.

Existen dos emuladores de uso comun, Wine (libre) y WMware (comercial). De estos dos parece ser que WMware es de una calidad superior y que funcione sin grandes problemas.

Podriamos aconsejar el no usar emuladores para evitar problemas. A no ser que tengas necesidad de usar una aplicacion muy especial, existen versiones de programas con caracteristicas similares a los de Windows, nativos de Linux, con lo que probablemente te funcionen mejor, mas rapido y no te consuman tantos recursos. La eleccion es vuestra ;-)

2.9 ¿Qué tengo que saber antes de instalar linux?

Introducción

Si nos ceñimos a esta FAQ, en la sección 1 se proporciona una información de carácter general acerca de qué es Linux, su historia, características,.... En la segunda sección se añaden otros aspectos fundamentales como ¿Para qué quiero linux? ¿Qué espacio y memoria necesito? ¿Funcionará en mi equipo? (si lo tengo) ¿Cómo lo consigo? ¿Qué distribución es la que más me conviene? etc. Lo que se ha abordado en esta sección son los datos relativos al sistema donde vamos a instalar linux. Muchos usuarios desconocen por completo qué es lo que están usando: no saben que placa base tienen, ni que tipo de memoria, qué tarjeta de vídeo, si su ratón es serie o ps2, ...y por supuesto otras más sutiles como el refresco vertical de la pantalla de su monitor. Es necesario conocer el tipo de hardware de nuestro sistema informático para ver si es compatible con la distribución de linux que vamos a instalar.

Hay un par de libros de la editorial *Anaya* que son excelentes para prepararse:

- Patrick Volkerding, Kevin Reichard y Eric Foster-Johnsos, *Instalación y configuración de Linux*, Anaya, Madrid, 1999
- Naba Barkakati, *RedHat Linux a fondo*, Anaya, Madrid,1999

Antes de instalar Linux hay que preparar el PC para la instalación. Puede que ya dispongamos de un PC con algún sistema operativo instalado como MS-DOS, Windows o OS/2 o puede ocurrir que vayamos a comprarnos un PC nuevo para instalar Linux en él. En este último caso podremos elegir el hardware y los periféricos compatibles con linux. De todos modos en el apartado siguiente nos dedicaremos a considerar el hardware soportado por el sistema operativo linux.

Los componentes fundamentales

En la tabla adjunta se indican los requerimientos mínimos (M) y óptimos (O) para los componentes más típicos de un PC

- Procesador : Intel 80386 (M); Pentium, i486 y cualquier clónico de Intel (O)
- RAM: 8 MB (M); de 32 MB a 2 GB
- Tarjeta gráfica: VGA (M); SVGA o tarjeta soportada por las XFree86 (O)
- Disco duro: 125 MB (M); 500 MB o más (O)
- Bus: Todos (M = O)
- Lector CD-ROM: velocidad de x2 (M); x8 o mayor (O)
- Ratón: Microsoft, Logitech o compatible (M=O)
- Tarjeta de red: Ninguna (M); Cualquiera soportada (O)

A continuación veremos todo desde un punto de vista mucho más pormenorizado.

La Unidad Central de Procesos:
Linux funcioma bien con cualquier procesador a partir de i386, especialmente i486, Pentium, Pentium Pro Pentium II/III y los Intel compatibles AMD y Cyrix. No funcionará con ningún i286 o inferior porque no son de 32 bits (Aunque existen proyetos por ahi para hacerlo funcionar en 8086 /80286).

Buses:
Linux los admite todos: ISA, VLB, EISA, PCI, PS/2 y MCA (este último desde la versión 2.0.7). Hay dos placas base que no funcionan con linux: La Supermicro P5MMA y la Supermicro P5MMA98. Para que lo hagan es necesario descargar de la red BIOS adecuadas: *ftp.supermicro.com*mma9051.zip y ftp.supermicro.com/a98905.zip/

Memoria RAM:
Para que el sistema con X-windows funcione decentemente debería disponer de al menos 16 MB de RAM. Como máximo puede gestionar 2 GB de RAM, pero a partir de 1 GB, pueden aparecer grandes limitaciones en la celeridad si no se instala memoria caché adicional en el PC. Si la memoria es superior a 64 MB habrá de indicarse al sistema en el fichero de configuración de LILO `/etc/lilo.conf` como una línea (Esto no es necesario a partir de la serie 2.2.x del kernel); p.ej., si tenemos 128 MB pondremos

```
append="mem=128M"
```

Discos duros:
Linux funciona con cualquier disco duro compatible con el BIOS del PC. Para instalar linux en el PC, no vendría mal disponer de una partición de alrededor de 1 GB para el linux native. Si se usa un disco duro

de gran capacidad (que tenga más de 1024 cilindros), entonces habremos de situar LILO, el kernel y los archivos de configuración en los primeros 1023 cilindros. Realmente linux no trabaja con los discos directamente, sino con los controladores del disco. Linux funciona con todos los controladores IDE e IDE/ATAPI, ciertos controladores EIDE (que gestionan hasta cuatro dispositivos sean discos duros o CD-ROMs) y los controladores SCSI a excepción de los adaptadores SCSI de puerto paralelo.

Miscelánea:
La gran mayoría del hardware soportado por linux puede consultarse en las direcciones *http://users.bart.nl/~patrickr/hardware-howto/Hardware-HOWTO.html* `<http://users.bart.nl/~patrickr/hardware-howto/Hardware-HOWTO.html>`,*http://www.xfree86.org/* `<http://www.xfree86.org/>` y *http://linuxhardware.net/* `<http://linuxhardware.net/>`

Todo sobre controladores EIDE, SCSI, Lectores y grabadores CD-ROM, Unidades de cinta, Unidades extraíbles, ratones (trackballs y touchpads), tarjetas de red (Ethernet, RDSI, Token-Ring, ARCnet...), tarjetas gráficas incluídas en la última lista de XFree86, tarjetas de sonido, dispositivos PCMCIA, etc. , se encuentra en estas direcciones y deben consultarse para evitar sorpresas.

El peligro de Windows: Impresoras y modems

Existe un peligro en lo que respecta a ciertas impresoras y modems llamados winprinters y winmodems que no funcionan en linux.

Hay dos tipos de impresoras que definitivamente no funcionan con linux: todas las impresoras `Hewlett-Packard Desjet 820xx` y la `Sharp JX-9210` que solo funcionan con Windows 95. El resto son compatibles con linux, aunque en ciertos casos hace falta descargar controladores (Canon BJC600/800; HP Deskjet 500/.../855C) o programas que mejoran la calidad de impresión (como el free-lj4 para la serie HP Laserjet 4)

El problema de los winmodems es que parte del hardware es sustituido por software y ese software solamente es de Windows, con lo cual no pueden funcionar en linux. Lo malo es que al comprar un modem nos venden un winmodem sin avisar. Si tenemos uno de ellos lo mejor es reemplazarlo por otro que funcione en linux. Para ello lo mejor es consultar en la dirección *http://www.o2.net/~gromitkc/winmodem.html* `<http://www.o2.net/~gromitkc/winmodem.html>` para escoger uno adecuado.

Antes de la instalación

Antes de comenzar la instalación, es necesario recoger la información necesaria para llevar a buen término el proceso:

- Controlador marca y modelo del CD-ROM
- Marca y modelo de la tarjeta ethernet
- Modelo, fabricante y puerto usado por el ratón
- Modelo Chipset y fabricante de la tarjeta gráfica
- Servidor X compatible con la tarjeta gráfica
- Memoria de la tarjeta gráfica
- Modelo, fabricante y datos de la tarjeta de sonido
- Intervalos de trabajo de barrido horizontal (kHz) y de refresco vertical (Hz) del Monitor.
- Fabricante, modelo y puerto serie usado por el modem
- Fabricante y modelo de impresora
- Nombre del computador

- Nombre del dominio de red
- Dirección IP asignada al computador
- Dirección de red de area local
- Dirección de la máscara de red
- Dirección de transmisión de la red
- Dirección del encaminador
- Dirección del servidor de nombres

Todos estos datos serán necesarios para instalar convenientemente el hardware y configurar las X-windows.

Por último, antes de comenzar sería bueno haber leído un poco de linux o –mejor aún– tener un amigo que ya conozca el sistema y que se digne a acompañarnos durante la instalación y configuración de nuestra distribución.

3 Durante la instalacion

3.1 ¿Dónde instalo Linux?

Linux se puede instalar en cualquier disco que tengas en tu sistema y en cualquier particion del disco duro (Primaria o extendida).
No podras tener Linux en una particion compartida con otro sistema operativo, Linux necesita su propia particion/es para funcionar.

3.2 ¿Qué es una particion? ¿Cómo creo una particion?

Particionar el disco duro es una manera de dividir el disco fisico en varios discos logicos. O lo que es lo mismo, al particionar un disco, dividimos el disco en varias particiones independientes unas de otras, creando la ilusion de que tenemos diferentes discos, cuando en realidad lo que tenemos es un solo disco fisico dividido en partes.

Una particion es una de estas partes (divisiones) del disco.

Existen dos clases de particiones: primarias y extendidas. En un disco solo podras tener como maximo 4 particiones primaria y 1 extendida. En la particion extendida se podran definir todas (bueno tambien existe un limite, pero es alto) las unidades logicas que queramos. Con este sistema podemos tener una gran cantidad de particiones en nuestro disco.

Cualquier disco que tengamos en nuestro ordenador tiene al menos una particion primaria, que en la mayoria de los casos tiene un tamano equivalente al total del disco.

Unos ejemplos aclararan las cosas:

- Un disco de 1Gb con una sola particion, tendra una particion primaria de 1Gb (total del disco).
- Ese mismo disco podria tener 4 particiones primarias de 0.25Gb cada una, dando la ilusion de que tenemos 4 discos duros de 0.25Gb en vez de un solo disco de 1Gb.
- Otra combinacion posible podria ser 4 particiones primarias de 0.10Gb y 1 extendida con 6 unidades logicas de 0.10Gb, en este caso pareceria que tenemos 10 discos duros de 0.10Gb cada uno.

Las combinaciones son multiples y variadas y dependeran de nuestros gustos y de lo que necesitemos.

Casi todos los sistemas operativos traen un programa con el que podemos crear, modificar, borrar las particiones de nuestro disco. En Ms-Dos/Windows de llama FDISK, este programa solo puede trabajar con particiones de Ms-Dos/Windows. En Linux tambien se llama FDISK (/sbin/fdisk), pero es un programa mas potente, capaz de trabajar y crear particiones tanto para Linux como otros sistemas operativos. Si vas a trabajar con Linux, es recomendable el uso del FDISK que viene con tu distribucion, para evitar problemas.

Al contrario que Ms-Dos, Windows, OS/2, las diferentes particiones en linux no se denominan C:, D:, E:, ..., etc, existe una denominacion propia:

Si los discos son IDE:

- **/dev/hda**: Disco duro IDE como master en el canal IDE 1.
- /dev/hda1: Particion primaria 1 en /dev/hda
- /dev/hda2: Particion primaria 2 en /dev/hda
- /dev/hda3: Particion primaria 3 en /dev/hda
- /dev/hda4: Particion primaria 4 en /dev/hda
- /dev/hda5: Particion extendida 1 en /dev/hda
- /dev/hda6: Particion extendida 2 en /dev/hda
- .....
- .....
- /dev/hda16: Particion extendida 16 en /dev/hda
- **/dev/hdb**: Disco duro IDE como esclavo en el canal IDE 1.
- /dev/hdb1: Particion primaria 1 en /dev/hdb
- ....
- ....
- **/dev/hdc**: Disco duro IDE como master en el canal IDE 2.
- /dev/hdc1: Particion primaria 1 en /dev/hdc
- ....
- ....
- **/dev/hdd**: Disco duro IDE como esclavo en el canal IDE 2.
- /dev/hdd1: Particion primaria 1 en /dev/hdd
- ....
- ....

Si los discos son SCSI:

- **/dev/sda**: Disco duro SCSI nr.1.
- /dev/sda1: Particion primaria 1 en /dev/sda

- ....
- ....
- **/dev/sdb**: Disco duro SCSI nr.2.
- /dev/sdb1: Particion primaria 1 en /dev/sdb
- ....
- ....

IMPORTANTE: Es muy importante saber lo que se esta haciendo cuando trabajeis con programas que modifican la tabla de particiones de un disco. Al cambiar la tabla de particiones de vuestro disco, se pierden los datos contenidos en las particiones afectadas. Realizar copias de seguridad de los datos que querais mantener antes de usar FDISK.

3.3 ¿Porque necesito diferentes particiones?

El particionar el disco, es simplemente una manera de organizar tu disco duro. Podras organizarlo con una sola particion o en varias. Es el usuario el que debera decidir cuantas particiones tendra su disco, y el tamano de las mismas, hay que recordar, que al menos hay que tener una particion primaria.

Desventajas de tener vuestro disco dividido en diferentes particiones.

- Ninguna

Ventajas en tener vuestro disco particionado en varias particiones:

- Si teneis un error/problema en una de ellas, las demas no se veran afectadas.
- Poder tener diferentes sistemas operativos en vuestra maquina, totalmente independientes unos de otros.
- Poder tener vuestros archivos de datos en particiones totalmente independientes.
- Poder borrar/cambiar el contenido de una particion, sin que esto afecte a las demas.

3.4 ¿Cuantas particiones necesito para Linux?

La respuesta rapida y facil es: recomendable al menos dos, una para el sistema/datos y otra para Swap. Usualmente se suelen tener tres, una para el sistema/programas (`/`), otra para los datos (`/home`) y otra para swap.

La respuesta larga y no tan facil es mas complicada de explicar: Todo dependera muchisimo del uso que se le vaya a dar al sistema.

Para sistemas que se utilicen de forma particular y por uno o pocos usuarios bastara con las dos/tres particiones antes mencionadas, esto evitara los problemas de saber que cantidad de espacio necesitan las diferentes particiones y el quedarnos sin espacio en alguna particion vital, mientras que nos sobra en otras.

Para sistemas servidores, con gran cantidad de servicios y usuarios es muy recomendable tener varias particiones/discos. Existe un documento (HOWTO: Multi Disk System Tuning) muy bueno y quizas complicado para el principiante que explica cuantas particiones y discos y que tamano deberian tener en funcion del uso que se le vaya a dar al sistema, lo podeis encontrar en *http://www.nyx.net/~sgjoen/disk.html* <http://www.nyx.net/~sgjoen/disk.html> o en cualquier servidor con documentacion Howto. Otro documento (HOWTO: Linux Partition) mas sencillo, se puede encontrar en *http://linux-es.uio.no/docs/HOWTO/mini/Partition* <http://linux-es.uio.no/docs/HOWTO/mini/Partition>.

3.5 ¿Qué es la Swap?

La swap es un espacio reservado en tu disco duro para poder usarse como una extension de memoria virtual de tu sistema. Es una tecnica utilizada desde hace tiempo para hacer creer a los programas que existe mas memoria RAM de la que en realidad existe. Es el propio sistema operativo el que se encarga de pasar datos a la swap cuando necesita mas espacio libre en la RAM y viceversa.

En Linux, la memoria total disponible por el sistema estara formada por la cantidad de memoria RAM instalada + la swap disponible. El acceso a la swap (disco duro) es mas lento que el acceso a la memoria RAM, por lo que si nuestro ordenador esta muy cargado de trabajo y hace un uso intensivo de la swap, la velocidad del sistema disminuira. Un uso muy intensivo y continuado de la swap es un indicativo de que necesitamos mas memoria en nuestro sistema para que funcione desahogado con el uso que le estamos dando.

En linux generalmente se usa como minimo una particion dedicada a swap (aunque tambien se puede tener un fichero swap).

3.6 ¿Cuanta Swap necesito?

Esta es otra pregunta que es dificil de contestar: Todo dependera del uso que se le vaya a dar al sistema y del espacio libre que tengamos.

Si vas a utilizar muchos programas a la vez y tienes poca memoria RAM, necesitaras mas swap, si tienes mucha RAM, no necesitaras tanta swap. Hay que recordar que un uso no intensivo de la swap es normal y no afectara mucho a la velocidad del sistema, pero como hemos dicho antes, un uso muy intensivo y continuado es un indicativo de que necesitamos mas memoria RAM.

No existe una formula magica para saber cuanto espacio deberiamos reservar para swap. Hay que recordar que la memoria total disponible en Linux es RAM + Swap. Como datos orientativos podriamos decir que como minimo, esta combinacion deberia ser de 32MB para sistemas que se utilicen en modo texto y de 64MB en adelante para sistemas que se utilicen en modo grafico.

Aqui teneis una serie de consejos sobre la swap:

- Es recomendable el tener siempre algo de swap configurada, incluso en sistemas con mucha memoria.
- En linux no se puede tener mas de 128MB en una particion swap, si necesitas mas de 128MB de swap, puedes tener mas de una particion dedicada a swap, por ejemplo, dos de 128MB
- Si tienes mas de un disco, instala la swap en el que trabaje mas rapido.
- Si tienes mas de un disco que se puedan acceder simultaneamente (por ej. dos SCSI, o dos IDE en diferentes canales IDE), se puede ganar algo de velocidad teniendo una particion swap en cada uno de estos discos.

Para un uso privado "normal-alto"del sistema, aqui teneis unos ejemplos orientativos, segun mi opinion, de la cantidad de swap recomendable:

- 16MB RAM + 64MB Swap
- 32MB RAM + 96MB Swap

- 64MB RAM + 64MB Swap
- 128MB RAM + 128MB Swap
- 256MB RAM + 128MB Swap

Como dato anecdotico, decir que existen servidores, en donde la cantidad de swap llega a ser de 256MB y hasta 512MB aunque yo tengo ordenadores con 16MB de RAM y 16MB de Swap como servidores de impresion que funcionan sin ningun problema.

En fin, cada uno debe de saber el uso que le va a dar a su sistema y analizar la cantidad de memoria necesaria, aunque si teneis espacio de sobra en el disco, mas vale tener mas Swap disponible que la necesaria, por lo que pueda ocurrir en un futuro, como dice el dicho, mas vale que sobre que no que falte.

3.7 No tengo sitio en mi disco duro, ¿Qué hago?

Si tienes todo el espacio de tu disco ocupado por otros sistemas operativos, deberas conseguir espacio para poder crear la particion/es donde instalar Linux. Existen tres maneras de hacer esto:

La primera, compra un disco duro he instalalo.

- *Ventajas*: Es la manera mas facil y segura de conseguir mas espacio.
- *Desventajas*: Es la mas cara ;-), tienes que instalar un nuevo disco duro.

La segunda, es borrar todas las particiones de tu disco duro, con lo que perderas toda la informacion que contiene, y empezar desde cero a definir las diferentes particiones del disco. Asi podras definir las particiones para Linux y otros sistemas operativos si vas a tenerlos.

- *Ventajas*: Es la manera mas facil y barata de conseguir mas espacio, no necesitas comprar otro disco.
- *Desventajas*: Pierdes la informacion que ya tengas en el disco, mas trabajo, ya que si tienes mas de un sistema operativo, tienes que instalarlos tambien aparte de Linux.

La tercera, existen unos programas que permiten cambiar la tabla de particiones del disco, sin perder los datos de las mismas. Si tienes por ejemplo una particion de 1GB con 500MB libres, puedes utilizar uno de estos programas para "robarle"espacio a la particion y crear una nueva particion, teniendo al final por ej. una particion de 600MB con la informacion de la particion original y una nueva particion con el resto de espacio liberado.

- *Ventajas*: Liberas espacio sin perder datos, no tienes que instalar de nuevo la informacion/sistema operativo contenido en la particion original.
- *Desventajas*: Estas jugando con la tabla de particiones del disco, hay que saber muy bien lo que se hace y como utilizar estos programas "ladrones de espacio". Si algo va mal, puedes perder la informacion que tenias enm la particion original y deberas empezar de cero como en el caso segundo.

Existen dos programas "ladrones de espacio"que se suelen utilizar normalmente en el caso tercero:

- **Partition Magic**: Pagina web principal *http://www.powerquest.com/root/intnl/spanish.html* `<http://www.powerquest.com/root/intnl/spanish.html>`
- **FIPS**: Pagina web principal *http://bmrc.berkeley.edu/people/chaffee/fips/fips.html* `<http://bmrc.berkeley.edu/people/chaffee/fips/fips.html>`

3.8 ¿Cómo configurar el arranque de Linux?

La manera mas facil de arrancar Linux es con LILO. Pasate por la seccion 4 (Sobre LILO) para mas informacion.

3.9 ¿Cómo utilizar LoadLin para el arranque de Linux?

A veces ciertos programas o "sistemas operativos"están diseñados para modificar el *Master Boot Record* (MBR) del ordenador sin tener en cuenta los contenidos actuales del mismo, lo que suele redundar en la eliminación de LILO y la imposibilidad del usuario de entrar en Linux. Para solucionar estos problemas, o por si simplemente deseamos entrar en Linux desde msdos, está disponible el programa LOADLIN.

Loadlin es un programa de MSDOS que podemos encontrar en el directorio dosutils de los CDs de Linux (en la propia distribución). Este programa permite arrancar Linux desde MSDOS a partir de él mismo, un fichero kernel de Linux (como vmlinuz, bzImage o zImage) y una partición Linux. Mediante él podemos hacernos un disco de seguridad para arrancar Linux cuando LILO sea borrado por otros S.O.:

Creamos un directorio `a:\loadlin` y copiamos alli el ejecutable `Loadlin.exe`. Copiamos también cualquier núcleo de linux que tengamos disponible (por ejemplo `\vmlinuz` o `\boot\bzImage`) al mismo diskette.

Cuando necesitemos arrancar desde el diskette, arrancamos en msdos y ejecutamos lo siguiente:

Formato de uso:

```
loadlin kernel root=particion
```

En nuestro caso:

```
a:
loadlin bzImage root=/dev/hda4
```

(cambiando cada cosa por su nombre correcto de fichero de kernel y partición donde tengamos instalado Linux). Esto lo podemos escribir en el mismo diskette en un fichero linux.bat, por ejemplo. Tras ejecutarlo, Linux arrancará normalmente con dicho kernel y con los datos de la partición indicada, lo cual permite restaurar el MBR original (con LILO) ejecutando de nuevo LILO (mediante `/sbin/lilo`, o `/sbin/lilo -v`).

4 Sobre LILO

4.1 ¿Qué es LILO y para qué sirve?

LILO, literalmente, *el cargador de linux* (LInux LOader) es un gestor de arranque muy versátil que permite iniciar un sistema operativo (SO) cargando el sector de arranque de una partición del disco duro (o de un disquete).

Esto lo pueden hacer otros gestores de botado, pero sólo LILO puede cargar el kernel de linux y arrancar linux presentando además la posibilidad de pasar parámetros al kernel mediante líneas de comando.

LILO tiene una serie de componentes que se cargan de manera secuencial. En primer lugar, se carga la primera parte (el comienzo del código) del cargador en el sector de arranque. Cuando este primer tramo del cargador se ha iniciado, aparece una "L"en el monitor. Entonces, el resto (la mayor parte del código) del cargador (que junto con la primera parte del código se encuentra en el fichero `/boot/boot.b`) procede a iniciarse. Justo antes de que se cargue este segundo tramo de código, se presenta "I"en la pantalla. Si todo ha ido bien, después de haberse cargado, la siguiente letra de LILO, "L", aparecerá en la pantalla. Después de esto, se produce la carga de la tabla de descriptores a partir del fichero `/boot.map`, donde se establecerán las ubicaciones de los posibles kernels de linux así como los sectores de arranque correspondientes a otros SOs que también gestione LILO. Por último, se carga el SO seleccionado, que en principio será el kernel de linux. Si todo ha funcionado sin problemas, aparecerá una "O"en el monitor y se habrá escrito en la pantalla la palabra LILO.

Si hubieren problemas en el proceso de arranque la presencia o ausencia de las letras de LILO junto con algunos otros mensajes serán claves para diagnosticar la causa del fallo.

4.2 ¿Dónde se instala LILO?

Dependiendo de las necesidades que tengamos, podremos instalar LILO:

- En el *sector de arranque* de un disquete (floppy) (`/dev/fd0`)
- En el *sector de arranque* (superbloque) de una partición raíz de linux (primaria o extendida) (`/dev/hda1`, p.ej. para (E)IDE ó `/dev/sda1` p.ej. para SCSI)
- En el *MBR* (Master Boot Record) o *Sector Maestro de Arranque* (primer sector) del primer disco duro (`/dev/hda`, p.ej. para (E)IDE ó `/dev/sda` p.ej. para SCSI)

La opción de instalar LILO en un disquete ("de arranque") es el procedimiento más seguro, pero también el más lento de cargar linux en nuestra máquina.

Si solamente tenemos linux en nuestro ordenador, la opción idónea sería instalar LILO en el MBR del primer disco duro. Si vamos a disponer de otros SOs coexistiendo en nuestra máquina (MS-DOS, Windows), también podemos escoger la instalación de LILO en el MBR, y gestionar con él la carga de los SOs. Pero con esto hay que tener ciertas precauciones.

Si ya tenemos un SO en una partición, p. ej. Windows 95, cuando instalemos LILO en el MBR, sobreescribiremos dicho sector y ya no podremos arrancar Windows 95, si no lo gestiona el propio LILO. Si tenemos linux instalado con LILO en el MBR y después instalamos p. ej., Windows 95, se borrará LILO del MBR y no podremos arrancar linux. En tal caso, para poder usar LILO como gestor común, habría que cargar linux con un disquete de arranque y reinstalar, desde linux LILO en el MBR. Por lo tanto, si vamos a meter varios SOs en nuestra máquina, es mejor instalar primero los otros y luego linux, usando después LILO como gestor de arranque de todos los SOs.

Si nos planteamos iniciar los diversos SOs, incluído linux desde otro gestor de arranque, como por ejemplo el moderno BootMagic del PartitionMagic 4.0 (Englobado ya en El Caldera OpenLinux 2.2), la mejor opción sería escribir LILO en el sector de arranque de la partición linux nativa correspondiente. En tal caso, cuando instalemos linux, deberemos marcar dicha partición raíz como activa (usando el comando ª"de `fdisk` o el "b"de `cfdisk`).

4.3 ¿Qué limitaciones impone el BIOS a LILO?

Excepto en el caso de que LILO sea el gestor de otros SOs, será el BIOS u otro cargador el responsable de ejecutar el sector de arranque de la partición. El BIOS (Basic Input Output System) cuando encendemos el ordenador, lleva a cabo el POST (Power-On Self Test) para comprobar los parámetros de la configuración de hardware que se encuentran permanentemente salvados en el chip CMOS (Complementary Metal Oxyde Semiconductor) RAM (continuamente alimentado por una batería aunque el ordenador se apague): memoria instalada, disqueteras, discos duros, cd-roms,...

Despues del test, BIOS muestra en la pantalla los valores de los parámetros de hardware y procede a arrancar el SO. En el caso de un disco duro toma el MBR del disco primario y de él se cargará el programa gestor de arranque. Antaño, el MBR contenía el sistema de arranque del único sistema operativo instalado. Hoy día como es posible cargar más de un SO, para arrancar, hace falta un gestor de arranque múltiple, como por ejemplo LILO. Por lo tanto, LILO está limitado por las opciones del BIOS. Además, LILO depende del BIOS para cargar los ficheros /boot/boot.b; /boot.map; los diversos kernels; los sectores de arranque de los otros Sos y el mensaje de arranque de LILO si se ha definido.

Las restricciones más importantes que afectan LILO eran en el caso de BIOS antiguos la limitación a los dos primeros discos duros y la imposibilidad de acceder más allá de 1024 cilindros por disco, lo cual se vulneraba en cuanto el disco duro superaba los 504 Mbytes, por lo que convenía tener la partición raíz que contuviera los kernels en las primeras posiciones de la tabla. Los BIOS de las modernas placas base ya soportan hasta cuatro dispositivos (discos duros o cdroms). Así, cuando la partición de arranque de linux se encuentra en el segundo, tercero o cuarto disco duro, LILO imprime un mensaje de precaución pero continúa. El límite de los 1024 cilindros no ha cambiado con los nuevos BIOS, pero los controladores de disco con soporte de LBA (Logical Block Address)o LARGE(IDEs con más de 1024 cilindros, la mayoría de EIDEs y todos los SCSI) pueden "traducirº "remapear"la geometría del disco en otra equivalente de manera que parezca que el sistema tiene menos de los 1024 cilindros y se puedan gestionar hasta casi 8 Gbytes. Si la información acerca de la geometría del disco que hay en el menú del BIOS (remapeada con p.ej. LBA) no es la misma (en cuanto a cilindros/cabezas/sectores) que la que nos da fdisk, entonces se producirá un problema de "geometry mismatch"(no equivalencia de geometría) y LILO abortará el arranque a menos que se utilice la opción "linear"en la configuración de LILO.

4.4 ¿Cómo se configura e instala LILO?

La configuración de LILO se hace en `/etc/lilo.conf`, pero antes que editar y escribir manualmente la configuración en este archivo, existen interfaces o programas capaces de preparar una configuración conveniente para el usuario, de manera que solamente en el caso de incorporar opciones avanzadas tenga que editar y modificar a mano el archivo.

En primer lugar existe un programa llamado "Quickinst"que acompaña la distribución original del programa LILO. Se trata de un guión (shell script) que produce una configuración mínima de LILO que luego conviene actualizar. Por otra parte la mayor parte de las distribuciones proporcionan un asistente durante la instalación que ayuda a la configuración de LILO. RedHat en su "INSTALL"dedica una parte importante a la configuración de LILO. El"SETUP"de Slackware también lo permite. Distribuciones como Caldera OpenLinux o SuSE disponen de herramientas de instalación que permiten una cómoda configuración de LILO para el usuario (LISA y LIZARD para OpenLinux 1.3 y 2.2, y YAST para SuSE). Quizá Debian ha sido un poco más espartana (al menos hasta Hamm 2.0) en su utilidad de instalación para la configuración de LILO: se limita a escribir una configuración básica de LILO sin soporte para otros SOs ni otros kernels.

El fichero `/etc/lilo.conf` solo debería ser leído por el superusuario administrador, ya que podría contener contraseñas. En caso de duda, deberíamos hacer (como root)

```
chmod 600 /etc/lilo.conf
```

Escribiendo /etc/lilo.conf

El fichero `/etc/lilo.conf` comienza con una "global section"(sección general) seguida de una o más "system sections"(secciones para cada SO que LILO gestione) que comenzarán por image= (si es un kernel linux) u other= (si es otro SO). Los comentarios en el fichero comienzan con el símbolo de la almohadilla (#).

Consideremos a continuación las líneas más importantes de `/etc/lilo.conf`:

Seccion general

- `boot=`*`dispositivo`* indica el dispositivo donde se instalará LILO. *dispositivo* puede ser un disquete (`/dev/df0`), una partición (p.ej. `/dev/hdc2`) o el disco completo (p.ej `/dev/hda`) que corresponde a la instalación en el MBR
- `install=`*`ficheroarranque`* instala el fichero especificado como nuevo sector de arranque. Por defecto es `/boot/boot.b`
- `map=`*`ficheromapa`* establece la ubicación del fichero de mapeado. Por defecto es `/boot/map`.
- `prompt` ordena a LILO que muestre un indicador (boot:) al momento de iniciarse el arranque que permita seleccionar el SO a iniciar. Si en este punto pulsamos la tecla TAB, aparecerán los nombres dados a los correspondientes SOs.
- `timeout=`*`tiempo`* indica el tiempo en décimas de segundo que LILO debe mostrar el indicador anterior. Si pasado ese tiempo, no se indicase nada, LILO cargaría el primer SO configurado.

Seccion linux

- `image=`*`kernelimagen`* donde introducimos el fichero que contiene el kernel, incluyendo la ruta de directorio. Lo más normal es que sea `/vmlinuz` o `/boot/vmlinuz`
- `label=`*`nombre`* donde escribimos un nombre para el SO en cuestión, p. ej., "linux"para linux, pero para otros, puede ser "DOS", "win98"...como máximo pueden ser cadenas de 15 caracteres. Introduciendo ese nombre cuando LILO presenta el indicador (boot:) se selecciona el SO a iniciar.
- `root=`*`partición root`* donde indicamos el nombre de la partición linux nativa de nuestro sistema linux, p. ej., /dev/hdc2.
- `read-only` indica al kernel que ha de montar inicialmente la partición raíz en modo de sólo lectura. Es necesario indicar esta opción para la partición donde se montará el sistema de ficheros raíz.

Seccion otros SOs

- `other=`*`partición`* donde indicamos esta vez la partición de inicio del SO en cuestión, p.ej., `/dev/hda1`.
- `label=`*`nombre`* tal como se indicó antes, será el nombre con que etiquetemos dicho SO.
- `loader=`*`gestor de arranque`* Para cargar el SO, LILO construye un pseudo MBR en su fichero de mapeo. Esta opción especifica el fichero donde tomar el código del pseudo MBR. Por defecto el fichero es `/boot.chain.b`
- `table=`*`tabla`* establece el dispositivo fuente para la tabla de partición escrita en el pseudo MBR, usualmente `/dev/hda` o `/dev/sda`.

Otras opciones interesantes (solo algunas)

- `vga=`*`modo`* selecciona el modo VGA al inicio. Los modos válidos son "normal"(80x25), "ext"(extendido, 80x50) o ªsk"(preguntar).
- `linear` hace que las referencias a los sectores se escriban como direcciones lógicas en lugar de físicas. Se emplea cuando LILO no reconoce correctamente la geometría del disco duro (debido a un remapeado por parte del BIOS).
- `message=`*`ficheromensaje`* establece el fichero cuyo texto se mostraría durante el arranque. No debe tener más de 24 líneas.

- `append=`*parámetro* permite pasar parámetros y componentes de hardware al kernel como cadena de caracteres. Por ejemplo, si nuestra memoria RAM es superior a 64 Mbytes, por ejemplo, 128, debemos pasarle el dato al kernel durante el arranque mediante (no necesario a partir de la serie 2.2.x)

```
append="mem=128M"
```

- `delay=`*tiempo* establece el tiempo en décimas de segundo que LILO esperará antes de arrancar el primer SO. Generalmente se usa cuando se arranca un único SO, pues si no, la opción prompt es mucho mejor. Por defecto (sin la opción) LILO no espera y se inicia el SO inmediatamente.

Después de haber escrito el fichero `/etc/lilo.conf`, es necesario lanzar el "map installer"haciendo:

```
/sbin/lilo
```

con lo cual LILO escribe un backup del sector de arranque, escribe la primera parte de su código en él y crea un nuevo fichero de mapeado (/boot/map). LILO anuncia entonces por la pantalla los SOs que gestiona, añadiendo una estrella al que se arrancará por defecto. Hay que recalcar que si cualquiera de los componentes de LILO cambia o se modifica su configuración mediante /etc/lilo.conf, es necesario volver a reinstalar LILO con /sbin/lilo.

Configuración mínima de LILO y configuración para gestionar linux y otro sistema operativo (win95)

Consideremos el siguiente fichero `/etc/lilo.conf` únicamente para arrancar **linux**:

```
#
#Fichero de configuración /etc/lilo.conf
#
# Sección general
boot=/dev/hda
delay=100
vga=normal

# Partición linux

  image=/vmlinuz
  root=/dev/hda1
  label=linux
  read-only
```

Este ejemplo se interpreta de la siguiente forma:

LILO se va a instalar en el MBR del disco duro `/dev/hda`. El modo de video EGA es 80x25, el sistema linux nativo se encuentra en la primera partición del disco duro (`/dev/hda1`) y la imagen del kernel se encuentra en el fichero /vmlinuz. Despues de 10 segundos, arrancará el sistema linux, cuyo nombre es "linux". Si hacemos `/sbin/lilo`, entonces aparecerá en el monitor:

```
Added linux*
```

Veamos ahora otro fichero de configuración para gestionar linux y win95:

```
#
# Fichero de configuración /etc/lilo.conf
```

```
#
# Sección general

boot=/dev/hda
prompt
timeout=100
vga=normal

# Partición linux

  image=/vmlinuz
  root=/dev/hdc1
  label=linux
  read-only

# Partición Windows 95

  other=/dev/hda1
  label=win95
  table=/dev/hda
```

En este caso, LILO también está en el MBR del primer disco duro (`/dev/hda`) y hay dos sistemas operativos: linux en la partición `/dev/hdc1` y Windows 95 en la partición `/dev/hda1`. La tabla de partición para el gestor de arranque corresponde al dispositivo `/dev/hda`.

si hacemos `/sbin/lilo`, aparecerá el mensaje:

```
Added linux*
Added win95
```

lo que significa que LILO va a gestionar los dos sistemas operativos y que por defecto el primero en arrancar será linux.

Cuando reinicialicemos el sistema, si todo ha ido bien aparecerá la palabra LILO seguida del indicador de arranque

```
LILO boot:
```

y así permanécera durante unos 10 segundos, pasados los cuales si no hemos escrito el nombre de uno de los SOs, cargará por defecto linux.

Si antes del tiempo, pulsamos la tecla TAB, entonces aparecerán en pantalla los nombres de los SOs gestionados:

```
linux       win95
```

4.5 ¿Cómo desinstalo LILO?

Hay razones para desear desinstalar LILO. Una, desinstalar el sistema linux de nuestra máquina. Otra imperativa: porque algo va mal. LILO es un programa que, mal manejado, puede hacer al sistema no arrancable si se instala en el MBR. Mal configurado, impedirá el acceso al cualquier SO. En tales casos es indispensable disponer de disquetes de arranque para dichos SO. Por lo tanto es conveniente hacer el disco de arranque de Windows y el de arranque de linux (en el momento adecuado de la instalación del sistema) de manera que nos permita acceder via disquetera al sistema correspondiente en caso de imposibilidad de arrancar. Nótese que en el BIOS el arranque deberá comenzar por defecto en la disquetera A:.

Una vez que hemos arrancado el SO, hay muchas maneras de desinstalar LILO del MBR. Si nos encontramos en DOS o Windows, basta con ejecutar el comando MS-DOS

```
FDISK /MBR (ó SYS C: si C: es el disco duro en cuyo MBR se instaló LILO)
```

si estamos en OS/2

```
FDISK /NEWMBR
```

que escriben los primeros 446 bytes (el código de arranque) en el MBR y dejando las particiones ilesas.

si estamos en linux, basta con hacer

```
/sbin/lilo -u
```

para restaurar el MBR anterior a la instalación de LILO

Además de todo esto puede hacerse otra cosa: Cuando LILO sobreescribe el MBR, salva una copia de backup en `/boot/boot.xxxx`, donde `xxxx` es un número mágico del dispositivo donde se ha instalado. Veamos algunos ejemplos

```
Disco         /dev/zzz              xxxx
IDE primario  /dev/hda              0300
SCSI primario /dev/sda              0800
floppy        /dev/fd0              0200
```

así, si queremos desinstalar LILO del sector de arranque de alguno de estos dispositivos, bastará con hacer:

```
dd if=/boot/boot.xxxx of=/dev/zzz bs=446 count=1
```

Si LILO se ha instalado en una partición raíz, no hará no arrancable ningún otro sistema operativo. Únicamente, si está mal instalado, hará que no arranque linux. Si deseamos restaurar el sector de arranque de la partición /dev/yyyy (p.ej. /dev/hda1) podemos reescribir el sector de botado haciendo:

```
dd if=/dev/yyyy of=New-file bs=512 count=1
dd if=Backup-Date of=/dev/yyyy
```

si lo que queremos es desinstalar linux, borraremos la partición root con fdisk.

4.6 LILO no funciona bien. Diagnóstico de errores.

Cuando LILO se inicia, si todo ha ido bien, presenta en la pantalla las cuatro letras "LILO". Como ya vimos, cada letra correspondía a una acción. Si LILO no ha funcionado, las letras que presente en pantalla junto con ciertos mensajes, serán la clave para saber donde está el problema. Veamos los más comunes:

- **"L"error**: El primer tramo del código ha sido cargado pero no el segundo. Esto se debe generalmente a un error físico en el sector de botado o a un problema de geometría del disco.

- **"LI"**

  :El segundo tramo del código ha sido invocado, pero no ha podido iniciarse. Esto sucede cuando hay un error de geometría de disco o se ha movido `/boot/boot.b` sin reinstalar LILO (`/sbin/lilo`).

- **"LIL"**

  :El segundo tramo del código se ha iniciado pero no puede ubicar los datos que necesita en el fichero de mapeado. Esto puede deberse a un error físico del dispositivo de arranque o a un problema en la geometría del disco.

- **"LIL?"**

  :El segundo tramo del código se ha cargado en una dirección de memoria equivocada. Esto se debe a un error en la geometría del disco o cuando se ha movido `/boot/boot.b` sin reinstalar LILO (`/sbin/lilo`).

- **"LIL-"**

  :Los datos en el fichero de mapeado no son válidos. Las causas son las mismas que en el caso anterior.

La mayor parte de las veces, no se trata de un problema físico del dispositivo de botado, sino errores debidos a la no equivalencia entre la geometría del disco que fdisk presenta y que el BIOS ha remapeado. En tal caso el uso del parámetro "linear"en `/etc/lilo.conf` puede funcionar. Otras veces puede deberse a que se ha intentado instalar LILO en una partición lógica. Y otras veces a causas más sutiles. Por ejemplo: Yo arranco linux junto a otros sistemas operativos con el gestor de arranque de Partition Magic y tengo LILO en una partición raíz. Una vez hice un "resize"y cambié el tamaño de mi partición linux nativa porque necesitaba espacio para instalar nuevos programas. Al terminar e intentar arrancar linux de su partición, apareció en la pantalla "LI". Eso se debió a que después de haber hecho el cambio de tamaño, había movido `/boot/boot.b` y no había lanzado el map installer `/sbin/lilo`. Cogí mi disquete de arranque de linux, y cuando entré ejecuté la orden y reinicié el sistema. Todo funcionó.

5 Kernel / Nucleo

5.1 ¿Qué es el kernel?

El kernel o nucleo de linux se podria definir como el corazon de este sistema operativo. Es el encargado de que el software y el hardware de tu ordenador puedan trabajar juntos.

Las *funciones mas importantes* del mismo, aunque no las unicas, son:

- Administracion de la memoria, para todos los programas en ejecucion.
- Administracion del tiempo de procesador, que estos programas en ejecucion utilizan.
- Es el encargado de que podamos acceder a los perifericos/elementos de nuestro ordenador de una manera comoda.

Existen dos *versiones* del Linux kernel:

- *Version de produccion*: La version de produccion, es la version estable hasta el momento. Esta version es el resultado final de las versiones de desarrollo o experimentales.

Cuando el equipo de desarrollo del nucleo experimental, decide que ha conseguido un nucleo estable y con la suficiente calidad, se lanza una nueva version de producion o estable. Esta version es la que se deberia utilizar para un uso normal del sistema, ya que son las versiones consideradas mas estables y libres de fallos en el momento de su lanzamiento.

- *Version de desarrollo*: Esta version es experimental y es la que utilizan los desarrolladores para programar, comprobar y verificar nuevas caracteristicas, correcciones, etc. Estos nucleos suelen ser inestables y no se deberian usar, a no ser que sepas lo que haces.

Como *interpretar los numeros de las versiones*:

Las versiones del nucleo se numeran con 3 numeros, de la siguiente forma: `XX.YY.ZZ`

- `XX`: Indica la serie principal del nucleo. Hasta el momento solo existen la 1 y 2. Este numero cambia cuando la manera de funcionamiento del nucleo ha sufrido un cambio muy importante.
- `YY`: Indica si la version es de desarrollo o de produccion. Un numero impar, significa que es de desarrollo, uno par,que es de produccion.
- `ZZ`: Indica nuevas versiones dentro de una version, en las que lo unico que se ha modificado, son fallos de programacion /bugs.

Unos *ejemplos* nos ayudaran a entenderlo mejor:

- `ej1`: version del nucleo 2.0.0: Nucleo de la serie 2 (XX=2), version de produccion 0 (YY=0 par), primera version de 2.0 (ZZ=0)
- `ej2`: version del nucleo 2.0.1: Nucleo de la serie 2, version 0, en el que se han corregido errores de programacion presentes en la version 2.0.0 (ZZ=1)
- `ej3`: version del nucleo 2.1.100: version 100 del nucleo de desarrollo 2.1.

5.2 ¿Dónde consigo el kernel?

El nucleo se puede bajar de un gran numero de servidores en internet. El servidor principal es *http://www.kernel.org/* `<http://www.kernel.org/>` y la pagina de servidores espejos es *http://www.kernel.org/mirrors/* `<http://www.kernel.org/mirrors/>`.

Si tienes problemas accediendo a estas paginas, aqui tienes una copia en otro servidor *http://linux-es.uio.no/lista_kernel.html* `<http://linux-es.uio.no/lista_kernel.html>`

5.3 ¿Cómo se configura e instala el kernel?

Este es uno de los temas que asustan a los nuevos usuarios de Linux. Lo primero deciros que no hay razon para asustarse, la configuracion e instalacion de un nuevo nucleo en nuestro sistema es mas facil de lo suena.

Lo que si hay que hacer es tener claro una serie de cosas antes de ponernos a trabajar, para asi evitar problemas. A continuacion tienes una pequena guia sobre como configurar e instalar un nuevo nucleo en tu sistema.

Has decidido instalar un nuevo nucleo en tu sistema, que es lo que tenemos que hacer?

1. Bajarte la ultima version. De donde? Leete la 5.2 (seccion anterior). **NOTA**: Si vais a instalar un nucleo de la serie 2.2.x, teneis que tener en cuenta que algunas distribuciones no estan/estaban preparadas para hacer uso de esta serie. Si vuestra distribucion no es de las que vienen preparadas, teneis que actualizar una serie de paquetes/programas antes de instalar el nuevo nucleo (mas informacion en la documentacion que acompaña al nucleo).

2. Tener claro lo que vamos a hacer, leete el documento HOWTO sobre el nucleo (*Ingles* `<http://linux-es.uio.no/docs/HOWTO/Kernel-HOWTO>` / *Castellano* `<http://linux-es.uio.no/docs/HOWTO/translations/es/Kernel-Como>`)

3. Tener claro las opciones que tenemos que configurar, para poder utilizar el hardware de nuestro sistema, asi como las caracteristicas que queremos utilizar. Por ejemplo, si no utilizamos un dispositivo SCSI,

no tenemos que configurar nada en el apartado SCSI de nuestro nucleo. Asi nos ahorramos espacio y tiempo.

4. Entrar como root: `su root`

5. Copiar el archivo que te has bajado al directorio /usr/src: `cp linux-xx.yy.zz.tar.gz /usr/src/`

6. Descomprimirlo y desempaquetar: `tar -xvzpf linux-xx.yy.zz.tar.gz` **NOTA**: El archivo `linux-xx.yy.zz.tar.gz` se desempaquetara en el directorio `/usr/src/linux`. Si ya existe un directorio llamado linux en tu sistema, renombralo, p.ej: `mv linux linux-old` . En algunas distribuciones, linux es un enlace simbolico a `linux-x.y.z`, borra este enlace simbolico. Es importante que no exista ningun directorio/enlace simbolico llamado linux, antes de desempaquetar la nueva version. Si te has bajado el kernel comprimido con bzip2, tendras que descomprimirlo con `bunzip2 linux-xx.yy.zz.tar.bz2`

7. Entrar en /usr/src/linux: `cd /usr/src/linux`

8. Configurar el nucleo, esto se puede hacer de tres maneras diferentes:
   - `make config` (modo texto)
   - `make menuconfig` (modo texto con menus)
   - `make xconfig` (X-windows version)

9. Si teneis X-windows instalado, os recomiendo el ultimo comando, si no el segundo.

10. Os recomiendo que las opciones que vienen por defecto no las toqueis, si no sabeis lo que haceis, podeis pulsar `Help` en cada opcion para obtener una descripcion de la misma. Configurar las opciones que quereis tener en vuestro nuevo nucleo. Una vez terminada la configuracion, grabar los cambios y salir del programa de configuracion.

11. Una vez terminado el proceso de configuracion, tenemos que compilar nuestro nuevo nucleo. Para ello hay que hacer lo siguiente:
    `make dep`
    `make clean`
    `make bzImage`

12. Si en el proceso de configuracion, elegimos alguna opcion como modulo, tendremos que compilar/instalar dichos modulos:
    `make modules`
    `make modules_install` **NOTA**: No olvidar ejecutar como root el comando `depmod -a` la primera vez que arranqueis con vuestro nuevo nucleo, para computar las dependencias entre modulos.

13. Ya tenemos el nucleo y los modulos compilados, ahora tenemos que instalarlo. Casi todo el mundo utiliza LILO para arrancar el sistema, por ello explicare como instalarlo utilizando LILO. Todavia estamos en `/usr/src/linux` , ejecutar el comando `make install` , esto copiara el nucleo que acabamos de crear, a el directorio `/boot` de nuestro sistema, con el nombre `vmlinuz`., o como un enlace simbolico `vmlinuz -> vmlinuz-xx.yy.zz`

14. Ahora tenemos que configurar LILO para que reconozca el nuevo nucleo. Tendremos que modificar el fichero /etc/lilo.conf. Aqui teneis un ejemplo, del fichero `/etc/lilo.conf` antes de modificarlo:

```
boot=/dev/hda
prompt
timeout=50
image=/boot/vmlinuz-2.0.34
```

```
    label=linux
    root=/dev/hda1
    read-only
```

Y aqui como quedaria despues de la modificacion para que reconozca nuestro nuevo nucleo al arrancar:

```
boot=/dev/hda
prompt
timeout=50
image=/boot/vmlinuz
    label=nuevokernel
    root=/dev/hda1
    read-only
image=/boot/vmlinuz-2.0.34
    label=linux
    root=/dev/hda1
    read-only
```

15. Ahora solo tenemos que ejecutar el comando `/sbin/lilo` y arrancar el sistema de nuevo. Si tuviesemos algun problema con el nuevo nucleo, siempre podriamos arrancar con el antiguo escribiendo `linux` y pulsando `ENTER` cuando arrancamos y nos sale en pantalla `lilo:` De esta manera podemos entrar y ver que es lo que ha fallado. **NOTA**: Recordar que existen multitud de opciones para configurar LILO, y que los ejemplos anteriores, son ejemplos. Puede que vuestro sistema necesite diferentes parametros y opciones. Leeros los documentos HOWTOS sobre el nucleo y LILO antes de cambiar nada en vuestro sistema. Al final de esta pagina teneis enlaces a los mismos. Suerte y espero que tengais las cosas un poco mas claras.

5.4 ¿Qué son los parches (patchs)? ¿Cómo se instalan?

Seguramente habreis oido hablar de los parches para el nucleo, si no, aqui teneis informacion sobre los mismos.

Que son los parches y para que sirven?:

Un parche para el nucleo no es mas, que un fichero que solamente contiene informacion, sobre las lineas de codigo que han cambiado desde la version precedente del nucleo. De esta manera, solamente te tienes que bajar un fichero con los cambios, en vez del nucleo al completo. El ahorro en cantidad de Mb bajados es bastante considerable, sobre todo para aquellos que dependen del modem y no tiene una conexion buena a internet.

Algo a tener muy en cuenta si vais a actualizar el nucleo por medio de parches, en vez de bajaros el nucleo al completo, es que teneis que ir actualizando de version a version. Para que se entienda un poco mejor, aqui teneis un ejemplo:

Si teneis el nucleo 2.2.0 y vais a actualizarlo al 2.2.1, os podeis bajar el fichero patch-2.2.1.gz [70Kb] en vez, del nucleo 2.2.1 al completo [12.5Mb]. Pero si teneis el nucleo 2.2.0 y vais a actualizar al 2.2.4, NO os vale bajaros el fichero patch-2.2.4.gz nada mas, tendriais que bajaros el 2.2.1, 2.2.2, 2.2.3 y 2.2.4. Esto es porque los ficheros patch solamente contienen los cambios de version a version.

Si la diferencia entre la version que teneis y la que quereis instalar, es muy grande (p.ej: del 2.2.0 al 2.2.10), no os merece la pena actualizar por medio de parches, en este caso bajaros la version completa.

Que hacer con un fichero patch-XX.YY.ZZ.gz?:

Ya te has bajado el fichero patch (se pueden bajar del mismo subdirectorio donde esta la version completa), que necesitas para actualizar el nucleo, y ahora que haces?. Ahora, hay que aplicarlo al nucleo que tienes y compilar de nuevo. El procedimiento para actualizar el nucleo por medio de ficheros patch es el siguiente:

- Copia el fichero patch-XX.YY.ZZ.gz al directorio /usr/src : `cp patch-XX.YY.ZZ.gz /usr/src/`
- Cambia a este subdirectorio y descomprime el fichero: `cd /usr/src/; gunzip patch-XX.YY.ZZ.gz`
- Aplica el parche: `patch -s -p0 < patch-XX.YY.ZZ`
- La opcion -s hara que solo se impriman mensajes de error. Si no recibes ningun mensaje de error (como deberia de ser ;-)) solamente te queda entrar en /usr/src/linux: `cd /usr/src/linux`
- Y ejecutar `make clean, make xconfig, make dep, make bzImage`, el resto es igual que en la 5.3 (seccion anterior) a partir del punto 12)

Esto es todo lo que se me ocurre sobre los parches del nucleo.

5.5 ¿Algun consejo sobre el kernel?

Pregunta: Necesito actualizar el nucleo que utilizo, cada vez que una nueva version aparece? **Respuesta:** NO. Porque? La explicacion es la siguiente:

Cuando un nuevo nucleo aparece, puede ser por las siguientes causas:

- Nuevas caracteristicas se han anadido.
- Fallos de programacion se han corregido
- Fallos de seguridad se han corregido.
- Nuevo hardware es soportado.

Si las caracteristicas que se han anadido, no las vamos a utilizar, es evidente que no necesitamos actualizar. Si los fallos de programacion que se han corregido, afectan a caracteristicas/drivers que no utilizamos, no necesitamos actualizar. Si no utilizamos el nuevo hardware soportado, tampoco necesitamos actualizar.

De todas maneras es recomendable, actualizar de vez en cuando, sobre todo cuando se corrigen fallos de seguridad o cuando los cambios en el nuevo nucleo afectan a caracteristicas/funciones/hardware que utilicemos. El codigo esta ahi, libre y esperando a ser compilado en un nuevo ordenador, cada usuario debe de decidir cuando es hora de una actualizacion.

Pregunta: Soy nuevo en Linux y acabo de instalar una distribucion, como actualizo el nucleo? **Respuesta:** Te aconsejo que esperes un poquito. La distribucion que acabas de instalar (si es de las ultimas) viene con un nucleo de los ültimos", totalmente funcional y que te sirve sin problemas. Utiliza el sistema un tiempo, familiarizate con el nuevo sistema que acabas de instalar, y cuando comprendas un poco mas como funcionan las cosas, actualiza el nucleo. Un buen punto de partida para encontrar informacion sobre el nucleo, lo tienes en estas paginas.

5.6 Enlaces sobre el kernel

- *The LinuxKernel Archives* <http://www.kernel.org/>: Pagina principal/oficial sobre el nucleo.
- *Kernel Traffic* <http://www.kt.opensrc.org/>: Informacion sobre la lista de correo sobre el nucleo.
- *KernelNotes* <http://www.kernelnotes.org/>: Mucha informacion sobre nucleo.
- *LinuxMama* <http://www.linuxmama.com/>: Actualizaciones no oficiales.
- *Kernel Howto* <http://linux-es.uio.no/docs/HOWTO/Kernel-HOWTO> (Ingles)
- *Kernel Howto* <http://linux-es.uio.no/docs/HOWTO/translations/es/Kernel-Como> (Castellano)
- *LILO MiniHowto* <http://linux-es.uio.no/docs/HOWTO/mini/LILO> (Ingles)
- *Guia de programacion* <http://metalab.unc.edu/LDP/LDP/lkmpg/mpg.html>: de modulos para el nucleo (ingles)
- *The Linux Kernel* <http://metalab.unc.edu/LDP/LDP/tlk/tlk.html>: Libro (ingles)

6 X-windows / Entornos graficos

6.1 ¿Qué es X-windows?

El protocolo X-windows fue desarrollado a mediados de los años 80 como respuesta a la necesidad de un interfaz grafico transparente sobre todo para sistemas Unix.

X-windows es el encargado de visualizar la informacion de manera grafica y es totalmente independiente del sistema operativo (los sistemas Unix/Linux no necesitan de X-windows para funcionar, pudiendo trabajar en modo texto). La gran diferencia entre X-windows y la interfaz grafica de otros sistemas operativos es que X-windows distribuye el procesamiento de aplicaciones, especificando un enlace cliente-servidor. El cliente X especificara "Que hacerªl servidor X, que se encargara de "Como hacerlo".

Pero dejemonos de teoria y pasemos a un ejemplo practico. La gran ventaja de X-windows es que el servidor X de una aplicacion y el cliente X donde trabajamos no tienen porque estar en la misma maquina. Podemos estar utilizando X-windows en nuestra maquina, conectarnos a otra remota, ejecutar un programa en esta maquina remota y utilizar/ver este programa en nuestra maquina local. Todo esto independientemente de la plataforma/sistema operativo que el ordenanor remoto utilice. Como veis las posibilidades son muchas y potentes.

```
     Cliente X
         |
         |
       Xlib
         |
         |
    Servidor X
         |
         |
Sistema operativo
```

6.2 ¿Cómo configuro las X-windows?

Para utilizar el sistema en modo grafico es necesario instalar X-windows en tu sistema. Existen servidores comerciales y bajo licencia GNU, en estos apartados nos basaremos en el X-windows bajo licencia GNU, XFree, ya que es el que se incluye en todas las distribuciones de Linux.

Una vez instaladas las X-windows tendremos que configurarlas para que funcionen en nuestro sistema. El fichero de configuracion del las X-windows se llama XF86Config y generalmente se encuentra en el directorio `/etc/X11/`.

Unos datos que teneis que saber antes de configurar X-windows son:

- Datos tecnicos de la tarjeta grafica que utiliceis, o por lo menos el nombre de la tarjeta. El programa `/usr/X11R6/bin/SuperProbe` os puede ayudar a obtener informacion sobre vuestra tarjeta.
- Datos tecnicos del monitor que teneis, resoluciones/frecuencias de refresco verticales/horizontales a la que puede trabajar. Este punto es importante para evitar problemas posteriores con vuestro monitor.

Antes de lanzar X-windows mediante el comando startx debemos asegurarnos de la correcta configuración de los diferentes parámetros del mismo (teclado, ratón, tarjeta de vídeo, etc.). Hay diferentes métodos para hacer esto.

En primer lugar tenemos los programas de configuración semi-automática. Según la distribución de Linux que utilicemos tendremos a nuestra disposición una serie de programas que nos permitirán realizar la configuración de XWindow de una manera más o menos sencilla:

- **/usr/X11R6/bin/xf86config** - Todas las distribuciones: Programa en modo texto.
- **/usr/X11R6/bin/XF86Setup** - SuSe, Redhat, Slackware: Programa en modo grafico, tiene que arrancarse en una consola texto.
- **/usr/X11R6/bin/Xconfigurator** Redhat: Programa semi-grafico con menus.

Estos programas nos preguntarán cosas básicas sobre el teclado, el monitor, la videomemoria o la tarjeta de vídeo instalada (lo harán cuando no sean capaces de detectarlas por sí mismos). Debemos responder siempre la opción más acertada y en cuanto a monitores, nunca especificar características superiores de las que disponemos. Un caso muy habitual en tarjetas muy nuevas es no encontrar un soporte exacto para los últimos chipsets, problema que se suele corregir acudiendo al soporte de framebuffer del kernel o esperando a que quienes trabajan en XFree86 obtengan por parte de los fabricantes (poco colaboradores) la información sobre cómo programar sus tarjetas.

En ciertos casos también se nos puede preguntar si deseamos que el sistema arranque directamente en XWindow o incluso testear los diferentes modos de vídeo para ver que si son correctamente soportados.

Además para que estos programas realicen una correcta configuración el sistema debe estar ya correctamente configurado en cuanto a modo texto se refiere. Por ejemplo, Xconfigurator no activará el teclado en castellano para XWindow si no hemos especificado que el idioma en que queremos usar nuestro Linux es este mismo, mediante la orden

```
export LANG=es_ES
```

bien en la línea de comandos, o en el fichero `/etc/profile`. Cuando `Xconfigurator` detecta esta variable de entorno ya sabe qué lenguaje utiliza el usuario y por tanto configurará XWindow para el correcto uso del teclado en este lenguaje. Otras variables indican el tipo de ratón, etc. y suelen ser correctamente especificadas por el programa de instalación. En el caso de `XF86Setup` la configuración permite una mayor participación por parte del usuario, ya que en lugar de realizar una configuración automática proporciona una lista de opciones para cada apartado, de modo que si conocemos nuestro hardware es la mejor manera de configurar el sistema. Además, a la hora de elegir resoluciones de vídeo, este programa permite especificar cual es la resolución por defecto deseada mientras que `Xconfigurator` deja marcada la menor de ellas como "por defecto"(o la primera que seleccionemos).

El programa de configuración `xf86config` es el más antiguo de todos, y realiza la configuración mediante preguntas que se responderán con teclado. Nos preguntará la tarjeta de vídeo, monitor, rango de frecuencias del monitor, videomemoria y resoluciones de trabajo deseadas, así como el mapa del teclado a utilizar (es 102 = España 102 teclas).

Como siempre, el método más eficaz de configurar XWindow es directamente editando los ficheros de configuración.Y es que otro método de configuración básico es sencillamente editar el fichero `/etc/X11/XF86Config`, que es en realidad lo que hacen todos los programas comentados anteriormente. En este fichero la configuración está dividida en secciones y mediante cualquier editor de texto llano es posible cambiar cualquier parámetro de XWindow.

Este fichero se divide en varias categorías o secciones (todas ellas con más subopciones muy bien comentadas en dicho fichero de configuración):

- Sección Files: Indica los caminos hacia las fuentes o ficheros de colores.

```
RgbPath "/usr/X11R6/lib/X11/rgb"
FontPath   "/usr/X11R6/lib/X11/fonts/75dpi:unscaled"
(etc.)
```

- Sección ServerFlags: Opciones varias sobre el servidor.
- Sección KeyBoard:Configuración del teclado. Para teclados en castellano las opciones son similares a las siguientes:

```
Protocol       "Standard"
LeftAlt         Meta
RightAlt        Meta
ScrollLock      Compose
RightCtl        Control
XkbKeycodes     "xfree86"
XkbTypes        "default"
XkbCompat       "default"
XkbSymbols      "us(pc101)"
XkbGeometry     "pc"
XkbRules        "xfree86"
XkbModel        "pc101"
XkbLayout       "es"
```

- Sección Pointer: Configuración del ratón:

```
# Para un ratón microsoft mouse o un genius:
Protocol    "Microsoft"
Device      "/dev/mouse"
```

- Sección Monitor: Datos sobre el monitor (frecuencias).
- Sección Device: Indica el servidor X a utilizar: Por ejemplo para una tarjeta con chipset Mach64:

```
Section "Device"
    Identifier  "Mach64 GB"
    VendorName  "ATI"
    BoardName   "ATI 3d Rage II"
    VideoRam    8192
EndSection
```

- Sección Screen: Modos de vídeo seleccionados: Por ejemplo para activar 320x200, 640x480, 800x600 y 1024x768 tanto a 8 como a 16 bpp de color, siendo usado por defecto el modo 1024x768x16 (el primero que aparezca en la lista):

```
Section "Screen"
   Driver      "accel"
   Device      "Mach64 GB"
   Monitor     "My Monitor"
   DefaultColorDepth 16
   Subsection "Display"
       Depth       16
       Modes       "1024x768" "800x600" "640x480" "320x200"
       ViewPort    0 0
       Virtual     1024 768
   EndSubsection
   Subsection "Display"
       Depth       8
       Modes       "1024x768" "800x600" "640x480" "320x200"
       ViewPort    0 0
       Virtual     1024 768
   EndSubsection
EndSection
```

Una vez creado este fichero, entramos en XWindow mediante el comando startx.

Del entorno XWindow se puede salir en cualquier momento mediante la combinación de teclas `Ctrl+Alt+Borrar`, o bien seleccionando LOGOUT en el menú del gestor de ventanas que utilicemos.

Para volver a las consolas de texto bastará con utilizar la combinación de teclas `Ctrl+Alt+F1`, +F2, etc (hasta F6 en las instalaciones por defecto). Para volver al terminal gráfico se utiliza de F7 en adelante (`Ctrl+Alt+F7`, +F8, etc.).

Si durante la instalación (o en la línea apropiada del fichero `XF86Config`) seleccionamos varias resoluciones de vídeo, es posible cambiar entre ellas mediante las teclas `Ctrl+Alt+'+'` y `Ctrl+Alt+'-'` (los signos del teclado numérico).

Además, si por defecto marcamos 8bpp (o 16 bpp) en las opciones, nada nos impide lanzar posteriormente XWindow en la profundidad de color deseada:

```
[sromero@localhost]# startx -- -bpp 8
[sromero@localhost]# startx -- -bpp 16
etc.
```

También es posible establecer la profundidad de color por defecto mediante una línea DefaultColorDepth = 16 en el fichero de configuración XF86Config, sección Screen.

Por último, saber que podemos lanzar más de una sesión de XWindow simultáneamente (en las consolas accesibles desde `Ctrl+Alt+F8`, +F9, etc.), mediante:

```
[sromero@localhost]# startx -- :1
[sromero@localhost]# startx -- :2
etc.
```

6.3 ¿Cómo cambio la resolucion por defecto?

En el fichero `/etc/X11/XF86Config` en la sección screen correspondiente al servidor que se está utilizando actualmente, en la subsección Display correspondiente a la Depth en que estemos trabajando, buscamos la linea:

```
Modes "resolución_1" "resolución_2"...
```

Un ejemplo:

```
Modes       "1024x768" "800x600" "640x480" "320x200"
```

Entonces ponemos la resolución que queramos que sea por defecto de primera. Estas resoluciones puestas aquí son entre las que podremos cambiar si queremos al pulsar `Ctrl+ Alt+ '+'` ó `Ctrl+Alt+'-'`, una vez arrancado X-windows..

6.4 ¿Cómo cambio el escritorio virtual?

En el fichero `/etc/X11/XF86Config`, en la sección screen correspondiente al servidor que se está utilizando actualmente, en la subsection Display correspondiente a la Depth en que estemos trabajando, añadimos la línea:

```
Virtual tamaño_x tamaño_y  #en píxeles
```

Un ejemplo:

```
Virtual     1024 768
```

Hay que puntualizar que si existe una resolucion definida de `Modes` mayor que el escritorio virtual, esta no se podra utilizar. Resumiendo, no se puede tener una resolucion de trabajo mayor que el escritorio virtual.

6.5 ¿Cómo cambio el numero de colores por defecto?

Hay dos maneras:

Mediante la línea de comandos:

```
[user@localhost]# startx --bpp resolución  (dónde resolución es 8, 16,24 ó 32)
```

Mediante el fichero de configuración:

En el fichero `/etc/X11/XF86Config`, en la sección screen correspondiente al servidor que se está utilizando actualmente, añadimos la línea:

```
DefaultColorDepth depth_que_queramos_utilizar
```

Un ejemplo:

```
DefaultColorDepth 16
```

Un Depth con valor 8 significa que trabajaremos a 256 colores, uno con valor 32 truecolor. No todas las tarjetas graficas podran utilizar todos los Depth disponibles en todas las resoluciones, todo dependera de la tarjeta grafica y de la memoria que esta tenga.

6.6 ¿Cómo arranco directamente en X-windows?

En Linux es perfectamente posible pedir que el arranque del sistema se haga en modo gráfico, y que el login y password se introduzcan directamente en una ventana XWindow para la posterior carga del gestor de ventanas habitual que use dicho usuario.

Es decir, podremos identificarnos y aparecer directamente bajo X sin necesidad de ejecutar `startx`.

Para arrancar directamente en X Window (o no hacerlo) todo el proceso de configuración gira en torno a cambiar el runlevel (o nivel de ejecución en que arranca Linux).

El runlevel es, dicho de una manera sencilla, el modo en que arranca Linux. Por defecto el runlevel suele ser el 2 ó el 3, es decir, arranque en modo texto o consola. Para cada distribución suele haber una lista de runleves y sus significados, aunque casi se puede decir que son similares para todas ellas. Para Redhat, por ejemplo, la lista es la siguiente:

```
# Porción del fichero /etc/inittab
# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
```

Como puede verse, el arranque normal por defecto para que puedan acceder usuarios a Linux es 2 (si no tenemos red) o 3 si queremos usar servicios de red, siendo normalmente este el runlevel por defecto. Como puede verse, X11 tiene asignado el runlevel al 5. Si cambiamos el runlevel por defecto para que arranque en el 5, nos aparecerán directamente X Window.

Para cambiar este runlevel por defecto hay que editar el fichero `/etc/inittab`, y acudir a la siguiente línea:

```
id:3:initdefault:
```

El valor numerico antes de initdefault indica el runlevel de arranque por defecto. Si lo cambiamos de 3 a 5, la próxima vez que arranquemos Linux arrancará en X Window:

```
id:5:initdefault:
```

Para que el proceso se ejecute correctamente debe tener instalado un gestor de arranque en runlevel 5, que es el programa encargado de pedir el login+passwd y arrancar el gestor de ventanas adecuado. Puede elegir entre xdm (el que viene por defecto con XWindow), kdm (version de kde del mismo) y gdm (versión aportada por gnome). Cada una de ellas dispone de ciertos ficheros de configuración o ejecutables para cambiar el fondo del login, el lenguaje, gestor de ventanas, etc. Consulte los ficheros en los paquetes relacionados.

Haga "man runlevel"para mas información sobre runlevels.

En las últimas versiones de Redhat, el programa `Xconfigurator` le permite elegir si desea o no arrancar directamente en XWindow (él mismo modifica el runlevel por defecto) mediante una simple pregunta a la que usted debe responder SI (Si desea arrancar en runlevel 5) o no (si desea arrancar en runlevel 3). Esta pregunta se le suele realizar al finalizar la selección de resoluciones y antes de salir de Xconfigurator.

Una vez arranque en XWindow, puede volver a cualquier consola de texto mediante las teclas `Ctrl+Alt+F1, Ctrl+Alt+F2, etc.` (por defecto entre F1 y F6 tendrás 6 consolas de texto), y volver a XWindow en cualquier momento mediante `Ctrl+Alt+F7` en adelante.

En algunas distribuciones puede ser necesario indicarle a inittab dónde está el gestor de runlevel 5 que debe arrancar:

Para xdm:

```
#Run xdm in runlevel 5
x:5:respawn:/usr/X11R6/bin/wdm -nodaemon
```

Para gdm:

```
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Estas líneas suelen ser automáticamente incluidas en el fichero por los rpm/deb instaladores de xdm, kdm y gdm. Consulte en la ayuda de dichos programas para más información.

6.7 ¿Qué es un gestor de ventanas?

Un gestor de ventanas no es otra cosa que el conjunto de programas, ventanas, funcionalidades, ... que hacen posible que el usuario pueda interactuar con el sistema de forma grafica y no en modo texto.

Para usar un gestor de ventanas, hay que tener configurado un *servidor X*. Tambien hay que decir que el *gestor de ventanas* utilizado es totalmente independiente del *servidor X* utilizado.

```
  Gestor de ventanas
        |
        |
     Cliente X
        |
        |
       Xlib
        |
        |
    Servidor X
        |
        |
Sistema operativo
```

Al contrario que en otros sistemas operativos, en Linux no es necesario utilizar un *servidor X - gestor de ventanas* para usar el sistema. El sistema operativo y el conjunto *servidor X - gestor de ventanas* usado, son cosas totalmente diferentes, independientes entre si. Es mas, existen usuarios que trabajan en modo texto sin ningun problema y sin usar un interfaz grafico.

Existen numerosos y variados gestores de ventanas para Linux, unos mejores y otros mas desarrollados y estables. Es el usuario el que tiene que decidir que gestor satisface mejor sus necesidades, pudiendo incluso tener mas de uno instalado. Para aclarar un poco las cosas, podriamos decir que, si un ordenador es usado por varios usuarios, todos utilizaran el mismo *servidor X* pero no necesariamente el mismo *gestor de ventanas*.

6.8 ¿Cómo elijo el Gestor de ventanas que quiero arrancar?

Una vez configurado X se hace necesario decirle a XWindow qué gestor de ventanas debe arrancar. Dicho de una manera sencilla, XWindow es la conjunción de 3 componentes:

- El entorno de ventanas X Window: Es quien implementa las llamadas internas de XWindow, el protocolo X, etc., es decir, el sistema de comunicaciones entre aplicaciones.
- El servidor gráfico X. Es un programa (un fichero ejecutable) encargado de responder a las órdenes gráficas encargadas por el entorno de ventanas. Es el encargado de dibujar rectángulos, puntos de color o pixels, rellenar zonas, dibujar imágenes, etc. La manera de decirle a XWindow qué ejecutable es el que deseamos usar es crear un enlace simbólico llamado `/usr/X11R6/bin/X` que apunte al servidor que usemos:

  ```
  Ej: ln -s /usr/X11R6/bin/XF86-S3 /usr/X11R6/bin/X
  (le diría que utilice el servidor de las S3).
  ```

  Este enlace es creado por el programa de configuración cuando seleccionamos la tarjeta en la lista que nos presenta.
- El gestor de ventanas es el módulo encargado de decidir que aspecto, forma, funcionalidades, atajos de teclado, opciones y menúes debe tener XWindow.

Así, al dividir XWindow en estos 3 módulos se nos permite cambiar cualquiera de ellos para dotar a XWindow del aspecto o funcionalidades que deseemos. El entorno de ventanas no suele cambiarse nunca (es XFree86), y el servidor X tampoco, ya que éste último es siempre el ejecutable apropiado para manejar nuestra tarjeta gráfica (XFree86-SVGA, XFree86-S3, etc.), pero el último es totalmente seleccionable entre multitud de opciones y es lo que nos permite cambiar radicalmente el aspecto de XWindow. Cambiando el gestor de ventanas (o Window Manager) podemos cambiar prácticamente el funcionamiento completo de XWindow.

Gestores de ventanas hay muchos: kde, afterstep, gnome, WindowMaker, icewm, etc, y todos tienen sus respectivos ejecutables: startkde, afterstep, gnome-session, wmaker, icewm, etc. Toda esta explicación se ha dado por un motivo muy sencillo: la posibilidad de elegir el Gestor de Ventanas deseado es parte de la configuración activa de XWindow, y vamos a ver la manera de decirle a XWindow qué gestor deseamos utilizar.

Para ello, simplemente debemos ir al directorio HOME del usuario en cuestion (`/home/usuario`) o de root (`/root`) y editar (creándolo si no existe) el fichero de texto .xinitrc . En este fichero pondremos una línea que indicará el Gestor de Ventanas que deseamos utilizar, en el formato:

```
exec ejecutable_del_gestor_de_ventanas
```

Además podemos utilizar líneas de comentarios que comiencen por el carácter '#' y que XWindow ignorará:

Ejemplos de ficheros `.xinitrc`:

```
# utilizar afterstep
exec afterstep
```

otro ejemplo podría ser lanzar kde:

```
# utilizar KDE
exec startkde
```

Así, dependiendo del ejecutable que lancemos (startkde, gnome-session, icewm, wmaker, ctwm, fvwm2, wmaker, blackbox, etc.) podremos arrancar el WM deseado.

6.9 ¿Dónde consigo un nuevo gestor de ventanas?

Muchas distribuciones incluyen en sus CDes una serie de gestores de ventanas. Tambien os lo podeis bajar de sus respectivos servidores web:

- **AfterStep**: *http://www.afterstep.org* `<http://www.afterstep.org>`
- **Gnome**: *http://www.gnome.org/* `<http://www.gnome.org/>`
- **KDE**: *http://www.kde.org* `<http://www.kde.org>`
- **FVWM**: *http://www.hpc.uh.edu/fvwm/* `<http://www.hpc.uh.edu/fvwm/>`
- **Enlightenment**: *http://www.rasterman.com/* `<http://www.rasterman.com/>`
- **WindowMaker**: *http://www.windowmaker.org/* `<http://www.windowmaker.org/>`
- **mlvwm**: *http://www.bioele.nuee.nagoya-u.ac.jp/member/tak/mlvwm.html* `<http://www.bioele.nuee.nagoya-u.ac.jp/member/tak/mlvwm.html>`
- **TkDesk**: *http://people.mainz.netsurf.de/~bolik/tkdesk/* `<http://people.mainz.netsurf.de/~bolik/tkdesk/>`
- **eXode**: *http://www.simplicity.net/exode/* `<http://www.simplicity.net/exode/>`
- **wm2**: *http://www.all-day-breakfast.com/wm2/* `<http://www.all-day-breakfast.com/wm2/>`
- **icewm**: *http://www.kiss.uni-lj.si/~k4fr0235/icewm/* `<http://www.kiss.uni-lj.si/~k4fr0235/icewm/>`

7 Administracion

7.1 ¿Puedo trabajar normalmente como root?

Un consejo, solo utilizar la cuenta de root (administrador) para tareas de administracion. Para trabajar normalmente con el sistema NO hacerlo como root, ya que se tiene acceso completo a todo el sistema y es facil de estropear cosas si no se sabe lo que se hace.

Aqui teneis un ejemplo que le ocurrio al autor de esta seccion trabajando como root:

> Sólo utilizo la cuenta de root para administración del sistema, pero un día estuve demasiado tiempo trabajando con ella para intentar empaquetar un rpm, resulta que lo instalé mal y creé el directorio `/usr/src/redhat/usr/src/redhat/SRC/`, claro eso no era lo que quería, así que tecleé:
>
> ```
> root@asimovI src]$ cd ..
> root@asimovI /]$ rm -rf *
> ```
>
> Yo pensaba que estaba dentro del segundo src, pero estaba en el primero, menos mal que me di de cuenta pronto y pulsé `Ctrl+c` para abortar el comando de borrado mientras se estaba borrando el directorio `X11R6`, así que no corrompí por completo el sistema, sólo tuve que volver a instalar algunos paquetes. Pero ¿y si no me doy cuenta?, pues que no podría haber hecho nada ya que hubiese borrado todo el contenido de `/usr` no existirían librerias, comandos, etc, teniendo que haber instalado el sistema operativo de nuevo.

Despues de este ejemplo, volvemos a recomendar lo que dijimos al principio, utilizar la cuenta de root solo en los casos necesarios y siempre teniéndole un respeto al poder que da. Estar seguros al 100% de lo que haceis como root, para no estropear el sistema. Una buena costumbre es nunca hacer login con root, sino trabajar como un usuario normal y cuando se necesite ser root hacer un `su`, y tan pronto como deje de ser necesario tener el privilegio hacer un `exit`, para volver a trabajar como usuario normal.

7.2 ¿Cómo abro/cierro una nueva cuenta?

Aquí veremos como abrir y cerrar una cuenta de usuario.

Abrir una nueva cuenta.

El abrir una nueva cuenta, no es más que añadir una entrada en el archivo`/etc/passwd` del sistema. Una entrada en este archivo tiene la siguiente forma:

```
<user>:<encrip>:<UID>:<GID>:<informacion>:<home>:<shell>

user:        Es el nombre de usuario.
encrip:      Es la clave encriptada.
UID:         Es el número UID con el que nos identificará el kernel.
GID:         El numero de grupo principal GID al que pertenecemos.
información: Información varia, tal como nombre completo del usuario,
             número de teléfono, e-mail, etc.
home:        Es el directorio home del usuario, es decir, el directorio al que
             accedemos justo después de hacer entrar en el sistema.
shell:       Es el shell que se abrirá para ejecutar las órdenes que de el usuario.
```

Hay utilidades que hacen que añadir cuentas de usuario sea más fácil. Entre ellas están el comando `adduser` o `useradd` y `passwd`. Lo pasos para crear un usuario son:

```
[root@asimovI /root] adduser mario
[root@asimovI /root] passwd mario
```

Para más información hacer `man adduser`, sobre todo si se quiere poner caducidad a las claves.

Hay otras utilidades más vistosas y fáciles de usar e intuitivas ya que son bajo X-Windows, un ejemplo de éstas son en el `contol-panel` de redhat, el YaST en SuSE, el gestor de usuarios de Kde, etc.

Hay que aclarar que si el campo de la clave encriptada del fichero `/etc/passwd` no contiene ningun valor, no sera necesario la utilizacion de una clave para entrar en el sistema. Esto es totalmente desaconsejable por los problemas de seguridad que puede crear.

Sin embargo es útil, por ejemplo, cuando nos olvidamos de la clave de root, la solución es arrancar con un disket de rescate, luego montar la partición de linux, y borrar la clave encriptada de la entrada de root en el fichero `/etc/passwd`. Luego reiniciamos el equipo, entramos como root y ejecutamos `passwd root` para poner la nueva contraseña.

Borrar una cuenta de usuario.

El comando a utilizar es `userdel` el cual tiene la siguiente sintaxis:

```
userdel [-r][login]
```

Este comando borra el usuario *login*, y si se añade la opción `-r` también borra su directorio de usuario.

Para borrar una cuenta de usuario tambien se pueden utilizar las utilidades gráficas que comentamos antes, de una forma muy intuitiva.

7.3 Tengo un ejecutable que no quiere ejecutarse.

A todos nos a ocurrido alguna vez, que al intentar ejecutar un comando/programa hemos obtenido como respuesta `command not found`.

Esto significa que el shell no encontro el comando que queriamos ejecutar. Las causas mas comunes suelen ser:

- El ejecutable no está en un subdirectorio incluido en el $(PATH)./
  La solucion es escribir el ejecutable con la ruta completa o moverte al directorio de éste y escribir `./programa`, donde programa es el nombre del ejecutable.
- El comando/programa que habeis escrito no existe
- Faltan variables de entorno.
  Busca ficheros de documentacion del programa, o prueba `comando -help`, para ver si hace falta añadir variables de entorno, para el uso del mismo.

Otra respuesta que podemos obtener al intentar ejecutar un comando/programa es `Permission denied`.

El error es bien claro, el ejecutable existe pero no tenemos permiso para ejecutarlo, comprueba que lo podemos ejecutar, es decir, tiene una x en el grupo al que pertenezcamos: dueño, grupo, u otros. Mas informacion sobre permisos en la subseccion 7.11 (¿Cómo se cambian los permisos de ficheros y directorios?).

Por ultimo, podemos obtener un error en el que se nos informa que cierta librería no se encuentra.

Ejecuta el comando:

```
[asimov@asimovI bin] ldd comando
```

Esto mostrará todas las librerías compartidas que el comando/programa utiliza/necesita para funcionar.

Si no tenemos algunas, las tenemos que instalar, pero si las tenemos tenemos que comprobar que `/sbin/ld` (el cargador de librerías dinámicas) sabe donde están, para ello hemos de añadir una entrada en el fichero `/etc/ld.so.conf` que consiste en poner el directorio donde esta la libreria, luego ejecutamos `/sbin/ldconfig -D` para cerciorarnos que la lee.

7.4 ¿Cómo dar formato a un diskett en Linux?

El formato de diskettes en Linux depende del tipo de formato que se le desee dar al diskette (tipo MSDOS, tipo FAT32/VFAT, tipo Ext2 de Linux, minix, etc.). En general simplemente hay que usar uno de los comandos mkfs disponibles, que creara el sistema de ficheros pertinente:

Para formatear diskettes en formato Linux-ext2

```
mkfs.ext2 [opciones] /dev/fd0
```

o bien su equivalente:

```
mke2fs [opciones] /dev/fd0
```

El usuario que formate deberá tener permiso de escritura para la disketera (normalmente `/dev/fd0`). Para eso debe estar en el grupo de usuarios floppy o que `/dev/fd0` tenga permisos 666 (para todo el mundo).

Para formatear diskettes en formato msdos

```
mkfs.msdos [opciones] /dev/fd0
```

o bien:

```
mkdosfs [opciones] /dev/fd0
```

Otro comando disponible si tenemos instaladas las mtools es mformat:

```
mformat a:
```

Este ultimo comando formateara el diskett en formato ms-dos.

Por último, existen otros programas como fdformat o superformat (ver paquete fdutils) que tambien sirven para dar formato a diskettes.

NOTA: Formatos del tipo ext2 y ms-dos son los que se utilizan mas usualmente en linux. Hay que aclarar que un diskett con formato ext2 *No sera posible leerlo* bajo ms-dos o Windows, solamente bajo Linux. Un diskett con formato ms-dos podra ser leido indistintamente por ms-dos, windows o Linux (si tenemos soporte en el kernel para trabajar con este formato).

7.5 ¿Cómo puedo montar y desmontar unidades en Linux?

Montar un sistema de ficheros/dispositivo a nivel usuario no es más que hacerlo disponible en el árbol de directorios de nuestro sistema. Como ya sabeis, en Linux vemos todos los sistemas de ficheros/dispositivos en un sólo árbol de directorios, no existen letras a:, c:, etc., así nos da igual que el contenido de un directorio sea un sistema de ficheros msdos, vfat, ext2, otro que está en una máquina remota, etc. Esta abstracción tiene un inconveniente, hay que montarlo, es decir, indicarle al kernel de Linux que a través del directorio XXXX, accedemos al sistema de ficheros/dispositivo yyy. Esto se hace con el comando mount (`man mount`, para detalles).

Montar un sistemas de ficheros/dispositivo a nivel kernel, no es más que rellenar unas tablas de registro. Es decir, ver si el sistema de ficheros está soportado, es decir, si existe la tabla de funciones con las que manejarlo. Luego registrar estas funciones y enlazar el directorio al sistema de ficheros. Esto es más bien lo que hace la llamada al sistema mount, la cual es llamada mediante el comando del mismo nombre.

Los parametros necesarios para montar un sistema de ficheros son:

- *Tipo de sistema de ficheros*: Es necesario indicarle el tipo de sistemas de ficheros a registrar. Así sabrá que funciones utilizar.
- *Tipo de acceso*: Si es de sólo lectura, sólo escritura o ambos.
- *Dispositivo*: Indicar el dispositivo sobre el que está físicamente el sistema de ficheros,(p.e.: `/dev/hda1, /dev/hdb, /dev/sda1`,..., none si es el `/proc`, ya que se realiza sobre la memoria).
- *Directorio*: Indica el directorio sobre el que se va a montar, es decir, el directorio mediante el cual accederemos al sistema de archivos.

Ejemplo: Montar el cdrom en el directorio /dev/cdrom.

```
mount -t <sistema de ficheros> <Dispositivo> <Directorio>
mount -t iso9660 /dev/hdb /mnt/cdrom
```

Supongo que ya comprendereis el inconveniente de esto: Cada vez que se quiere acceder a un cdrom se ha de ejecutar el comando anterior, en vez de teclear `d:` (como en otros sistemas operativos). Pero la versatilidad y la flexibilidad que ofrece hace que no se tenga en cuenta ese inconveniente (como usuario accedo dentro de un directorio sin que me pase por la cabeza que ese fichero está al otro lado del planeta utilizando para ello el protocolo nfs).

Existen otras formas de reducir la línea de comandos, para ello (entre otras cosas) existe el fichero *fstab*. En él se indican los sistemas de ficheros sobre los que trabajamos normalmente: el sistema de ficheros en el que tenemos los directorios de linux, el /proc, la partición dos, el cdrom, y el floppy.

El fichero */etc/fstab* funciona de la siguiente manera:

Partimos de un ejemplo de contenido de `/etc/fstab`:

```
# <device> <mountpoint> <filesystemtype> <options>                        <dump> <fsckorder>

/dev/hda2  /            ext2            defaults                            1       1
/dev/hda3  /usr         ext2            defaults                            1       2
/dev/sda1  /home        ext2            defaults                            1       2
/dev/hdb   /mnt/cdrom   iso9660         user,noexec,nodev,nosuid,ro,noauto  0       0
/dev/fd0   /mnt/floppy  vfat            user,noexec,nodev,nosuid,rw,noauto  0       0
none       /proc        proc            defaults                            0       0
/dev/hda4  swap         swap            defaults                            0       0
/dev/hda1  /mnt/dos     vfat            exec,dev,suid,rw,auto               0       0
```

Con la informacion contenida en este fichero, el sistema haria lo siguiente al arrancar el sistema:

- La particion `/dev/hda1` se montaria en el subdirectorio `/mnt/dos`
- La particion `/dev/hda2` se montaria en el subdirectorio `/`
- La particion `/dev/hda3` se montaria en el subdirectorio `/usr`
- La particion `/dev/hda4` se montaria en el subdirectorio como `swap`
- La particion `/dev/sda1` se montaria en el subdirectorio `/home`
- `Proc` se montaria en el subdirectorio `/proc`
- El sistema tendria informacion sobre como montar un disket `/dev/fd0` y un CD-Rom `/dev/hdb`, aunque no los monta automaticamente al arrancar por haber definido la opcion `noauto`.

Veamos la explicación de los parametros usados en `/etc/fstab`:

En la columna de dispositivo se indica el dispositivo/particion a montar, en la punto de montaje se indica el directoria mediante el cual vamos a acceder al sistema de archivos. En la columna de tipo de sistema de ficheros se indica el sistema de ficheros que se usara sobre el dispositivo.

Las opciones significan lo siguiente:

- *user,nouser*: permite/no permite a un usuario ordinario montar el sistema de ficheros.
- *suid,nosuid*: Se requiere / no se requiere tener privilegios de superusuario .
- *auto/noauto*: Indica que sí/no se monta cuando hacemos mount -a.
- *defaults*: Aplica las opciones rw,suid,dev,exec,auto,nouser,async.
- *exec*: Permite la ejecución de binarios.
- *ro,rw*: Montar sólo lectura, lectura-escritura.

El fichero `/etc/fstab` de nuestro ejemplo nos permitiría hacer lo siguiente para montar un diskett o CD-Rom:

```
mount /mnt/cdrom
mount /mnt/floppy
```

Cuando dejamos de utilizar un sistema de ficheros, tenemos que darlo de baja (desmontarlo), para eliminar todas las referencias del kernel a él. El comando utilizado para esto es `umount`.

Para desmontar un sistema de ficheros/dispositivo:

```
umount <directorio|dispositvo>
```

ej:

```
umount /mnt/floppy
umount /dev/fd0
```

NOTA: Es necesario ejecutar el comando `umount` cada vez que cambiemos un diskett, ya que no siempre cuando se escribe en un dispositivo, se realiza automáticamente un volcado al dispositivo (por eso se pueden perder datos si se apaga el ordenador a lo bruto). Además si cambiamos el disket sin hacer umount, el kernel piensa que hay el sistema de ficheros anterior y cuando guarda alguna información lo hace con referencia al retirado, perdiéndose todo.

Este es un punto a tener en cuenta con el uso de disketts, ya que el mecanismo para retirar un diskett es mecanico, no existe posibilidad de impedir por medio de software que alguien saque un diskett de la disquetera. No ocurre lo mismo con el CD-Rom, ya que el sistema bloqueara la posibilidad de extraerlo cuando este montado.

7.6 ¿Cómo se crea un diskette de arranque para Linux?

Solo tenemos que ejecutar los siguientes comandos con un diskette vacío dentro de la diskettera:

```
[sromero@localhost]$ /sbin/mke2fs /dev/fd0
[sromero@localhost]$ [ -d /fd ] || mkdir /fd
[sromero@localhost]$ mount /dev/fd0 /fd
[sromero@localhost]$ cp /boot/boot.b /fd
[sromero@localhost]$ cp /vmlinuz /fd
[sromero@localhost]$ echo image=/fd/vmlinuz label=linux | /sbin/lilo -C - -b /dev/fd0 -i /fd/boot.b -c -m /fd/map
[sromero@localhost]$ umount /fd
[sromero@localhost]$rmdir /fd
```

Lo que hacen estos comandos es:

```
/sbin/mke2fs /dev/fd0    -> Formatea el diskette con formato ext2
[ -d /fd ] || mkdir /fd  -> Si no existe el directorio /fd lo crea
mount /dev/fd0 /fd       -> monta el diskette en dicho directorio.
cp /boot/boot.b /fd
cp /vmlinuz /fd          -> Copia el kernel de Linux y el archivo
                            de boot al diskette. Si nuestro kernel no
                            se llama vmlinuz (por ej bzImage o zImage)
                            o no está en el raíz debemos cambiar esto.
echo image=...           -> Activa el kernel en el diskette.
umount /fd               -> Desmontamos el disco
rmdir /fd                -> Borramos el dir /fd
```

Si alguno de los paths o nombres cambia los deberemos cambiar dentro del diskette.

7.7 ¿Cómo se utilizan los empaquetadores-des/compresores?

Los ficheros tar no son ficheros comprimidos, sino empaquetados. Tar es un empaquetador, es decir, es algo parecido a un compresor como arj o zip, pero sin compresión. Se dedica a incluir todos los ficheros juntos en el mismo archivo, preservando las estructuras de directorios y permisos de los mismos. Como veremos, lo podremos comprimir gracias al programa GZip.

Hay 2 operaciones básicas con tar: empaquetado y desempaquetado. Si estamos en un directorio y queremos empaquetar todos los ficheros de este directorio y los que cuelgan de él, basta con ejecutar la orden:

```
tar -cvf fichero.tar *

c = compress  (más bien, empaquetar)
v = verbose   (para que nos diga lo que hace)
f = file      (empaquetar en un fichero)
*             (empaquetar todos los ficheros, podría haber sido *.doc, etc.)
```

Si disponemos de un fichero .tar y queremos desempaquetarlo:

```
tar -xvf fichero.tar

x = eXtract   (desempaquetar).
```

También es posible listar los contenidos de un fichero .tar antes de desempaquetarlo, mediante la orden tar tvf fichero.tar .

Por otra parte, el ficheros con extensión gz son ficheros comprimidos. A diferencia de arj o zip, el contenido de un fichero GZ es un solo fichero, es decir, cuando comprimimos fichero.txt con este compresor (llamado gzip) obtenemos un fichero.txt.gz de tamaño mucho menor. Con GZ no es posible empaquetar ficheros, es decir, la compresión se realiza a un sólo fichero.

Para comprimir un fichero con gz, se utiliza el comando:

```
gzip fichero
```

Para descomprimirlo:

```
gunzip fichero.gz
```

La combinación de tar y gz es lo que permite el tener multiples ficheros comprimidos en un sólo archivo. Es decir, si empaquetamos un directorio con tar y luego comprimimos ese archivo tar con gz, obtenemos un tar.gz comprimido con múltiples ficheros.

La compresión y descompresión es posible hacerla en 2 pasos (primero tar y luego usar gz) o bien usar el flag -z de tar para ello:

Compresión:

```
tar -cvzf fichero.tar.gz *
```

Descompresion:

```
tar -xvzf fichero.tar.gz
```

Otro formato que se ha puesto de moda es bzip2, con el mismo sistema de funcionamiento que Gzip, y cuyos nombres de ejecutable son `bzip2` (comprimir) y `bunzip2` (descomprimir). Este compresor obtiene mejor compresión que Gzip y su funcionamiento es igual de sencillo, aunque tarda mas en comprimir y utiliza mas recursos.

Estos compresores/descompresores/empaquetadores son una gran y libre alternativa a formatos comerciales como `zip, arj` y `rar`, tambien disponibles para Linux (comandos zip, unzip, rar y unarj).

Para descomprimir ficheros arj mediante unarj, simplemente hace falta ejecutar el comando `unarj x fichero.arj`. El compresor es shareware y se debe obtener en la Web de sus programadores.

Zip es el programa destinado a hacer Linux capaz de leer y escribir los ficheros en formato .zip (generados por `pkzip o winzip`): Para ello tenemos los comandos zip e unzip, que nos permitiran comprimir y descomprimir ficheros sueltos, directorios completos, directorios con recursividad, etc:

Para comprimir todos los ficheros de un directorio en un zip:

```
zip fichero.zip *
```

Para comprimir este directorio y todos los que cuelguen del mismo:

```
zip -r fichero.zip *
```

La descompresión se realiza mediante unzip:

```
unzip fichero.zip
```

El programa rar también es un buen compresor que podemos encontrar en diferentes formatos (rpm, deb, tar.gz) en Internet. Su uso es identico a la versión MSDOS:

Comprimir:

```
rar a fichero.rar *
```

Descomprimir:

```
rar x fichero
```

Para más información sobre cualquiera de los des/compresores basta con consultar la página man del mismo, mediante "man *comando*".

7.8 ¿Cómo instalo, desinstalo o actualizo paquetes rpm?

Los paquetes rpm son archivos que llevan incluidos dentro de ellos todos los ficheros que componen un determinado programa. Internamente están comprimidos, pero nosotros sólo debemos pensar en ellos en términos de Instalación, Actualización, Borrado y Consultas. Dentro del rpm van los ficheros del programa a instalar, su descripcion, a que directorios van a ir instalados, scripts de auto-configuración en algunos casos, etc.

La sintaxis de rpm es `rpm -accion nombre_del_paquete`

Acciones:

```
rpm -i  archivo   (instalar)
rpm -e  paquete   (desinstalar)
rpm -u  paquete   (actualizar)
rpm -qi paquete   (pedir info)
```

Ejemplos:

```
rpm -i  Par-1.50-1.i386.rpm
rpm -e  Par
rpm -u  Par
rpm -qi Par
```

Supongamos el fichero `programa-1.0.rpm` que no tenemos instalado y que acabamos de bajar de Internet. Procedemos a su instalación:

```
rpm -i programa-1.0.rpm
```

Tras eso el programa estará instalado en nuestro Linux y podremos ejecutarlo y usarlo normalmente. Tal vez nuestro problema es que no sabemos como se llama el ejecutable y los demás ficheros de configuración que le acompañan. Para solucionar eso hacemos una consulta (query) del paquete ya instalado:

```
rpm -ql programa
```

La acción -ql significa "query list", y nos mostrará en pantalla la lista de ficheros instalados de este programa y sus directorios destinos. Si por ejemplo deseamos ver sólo los fichero sean instalados en los directorios bin (los ejecutables) podemos hacer uso de grep, la herramienta de Linux que sólo nos mostrará aquellas líneas que contengan una cadena determinada:

```
rpm -ql programa | grep bin
```

Esto nos mostrará sólo los ficheros de "programa"que hayan sido instalados en directorios bin.

Si queremos saber que hace un paquete instalado, podemos verlo con la opción "query info"(-qi):

```
rpm -qi programa
```

Como ejemplo, veamos la salida para el paquete fetchmail de Linux:

```
Name        : fetchmail               Relocations: (not relocateable)
Version     : 5.0.0                   Vendor: Red Hat Software
Release     : 1
Install date: dom 30 may 1999 16:00:12 CEST
Group       : Applications/Internet
Size        : 565413
Packager    : Red Hat Software http://developer.redhat.com/bugzilla
Summary     : A remote mail retrieval and forwarding utility.
Description :
Fetchmail is a remote mail retrieval and forwarding utility intended
for use over on-demand TCP/IP links, like SLIP or PPP connections.
Fetchmail supports every remote-mail protocol currently in use on the
Internet (POP2, POP3, RPOP, APOP, KPOP, all IMAPs, ESMTP ETRN) for
retrieval.  Then Fetchmail forwards the mail through SMTP, so you can
read it through your normal mail client.
```

Si el programa no nos gusta, la deinstalación es muy sencilla:

```
rpm -e programa
```

Obviamente, no tenemos porqué instalar los programas para ver su contenido o información. Los podremos ver antes de la instalación insertando un comando p antes de la acción:

```
rpm -qpi fichero.rpm
rpm -qpl fichero.rpm
```

Si queremos ver la lista de RPMs instalados disponemos del comando "query all"(-qa):

```
rpm -qa
```

Para verlo en formato pausado, podemos usar una tubería:

```
rpm -qa | less
```

Es posible que tras un tiempo obtengamos la versión 2.0 del programa que ya disponemos instalado. En esto caso hay 2 opciones: bien eliminar el programa anterior (-e) e instalar este nuevo (-i), o, simplemente, actualizar el programa a la versión 2.0 con el comando -U (de Update):

```
rpm -U programa-2.0.rpm
```

Por último, si al tratar de instalar un paquete rpm aparece el siguiente error:

```
Data type X not supprted
```

Esto es debido a que nuestra versión de rpm es muy antigua, al menos más que el rpm que estamos tratando de instalar, y que este tiene algún tipo de compresión o elemento que nuestro rpm no entiende. Bastará entonces con actualizar nuestro ejecutable del RPM.

Cabe decir que también existen front-ends al programa rpm, es decir, programas en modo gráfico (o texto) que realizan las acciones del programa RPM mediante pulsaciones nuestras del ratón. Es el front-end el que se encarga de pasarle a RPM los parámetros correctos para que se realice la acción pedida por el usuario. Entre estos programas tenemos glint, gnorpm, purp, kpackage, xrpm, etc.

7.9 ¿Cómo instalo paquetes .deb de debian?

Al igual que el formato RPM, los .DEB encapsulan todos los ficheros y scripts vinculados a un determinado programa pero para la distribución Debian. Además de disponer de herramientas de instalación como dselect, se incluye la posibilidad de instalación manual mediante las diferentes opciones del programa `dpkg`:

Instalación de paquetes deb:

```
dpkg -i nombre_del_paquete.deb
```

Para hacer consultas:

```
dpkg -s nombre_del_paquete (admite comodines)
```

Para desinstalar:

```
dpkg -r nombre_del_paquete
```

El resto de opciones de dpkg puede verse con `man dpkg`.

7.10 ¿Cómo se convierten paquetes entre RPM, DEB y TGZ con alien?

Alien es un programa que permite convertir un paquete binario entre los formatos rpm (Redhat, Suse), deb (Debian) y tgz (Slackware):

Conversion a .DEB:

```
alien --to-deb paquete.elquesea
```

Conversion a .RPM:

```
alien --to-rpm paquete.elquesea
```

Conversion a .TGZ:

```
alien --to-tgz paquete.elquesea
```

Existen otros flags (como -i) que permiten directamente tras la conversión realizar la instalación del paquete generado. Alien realiza la conversión descomprimiendo el paquete original y generando el nuevo paquete, diciéndonos el lugar del árbol de directorios donde dejará el paquete recién generado, según la distribución (por ejemplo, en Redhat en `/usr/src/redhat/RPMS/`).

7.11 ¿Cómo se cambian los permisos de ficheros y directorios?

Lo primero que hay que decir es, que para conseguir toda la informacion sobre los comandos involucrados en el tema de permisos podeis consultar`man chmod, man chown` y `man chgrp`

Informacion de un fichero/directorio

Cuando obtienes informacion sobre un fichero/directorio, existen diferentes campos que te dicen que clase de permisos el fichero/directorio tiene.

```
Ej:

[user@localhost]# ls -l
-rwxr-x---   1 pepito depart1     4348 Nov 24 16:19 test
```

En la *primera columna* se pueden ver una serie de letras `-rwxr-x--`, estas letras nos dicen quien en el sistema, y que clases de permisos tiene el fichero `test`.

Estas letras estan agrupadas en tres grupos con tres posiciones cada uno, mas una primera posicion que nos dice de que clase de archivo se trata (los mas normales (d) directorios, o (-) archivos de datos). En nuestro ejemplo la primera posicion es (-) con lo cual el archivo test, es un archivo de datos (binario/ejecutable en este caso).

El primer grupo de tres (rwx en nuestro caso) nos dice que clase de permisos tiene el dueño del fichero (u)(user/owner)).
El segundo grupo de tres (r-x en nuestro caso) nos dice que clase de permisos tiene el grupo del fichero (g)(group).
Y el ultimo grupo de tres (— en nuestro caso) nos dice que clase de permisos tienen todos los demas usuarios del sistema sobre este fichero(o)(others).

```
r :significa permiso para leer
w :significa permiso para escribir
x :significa permiso para ejecutar
```

La *segunda columna* `pepito`, nos dice quien es el dueño del fichero,(pepito en este caso).
La *tercera columna* `depart1`, nos dice cual es el grupo del fichero (depart1 en este caso).
La *cuarta columna* `4348`, nos dice el tamaño del fichero.
La *quinta columna* `Nov 24 16:19`, nos dice cual es la fecha y hora de la ultima modificacion.
La *sexta columna* `test`, nos dice cual es el nombre del fichero/directorio.

Asi pues, el fichero `test` de nuestro ejemplo tiene los siguientes permisos:

pepito puede leer, escribir/modificar, y ejecutar el fichero test.
depart1 puede leer, y ejecutar pero no escribir/modificar.
Los *demas usuarios* no pueden hacer nada, ni leerlo, ni escribir/modificar, ni ejecutarlo.

Como cambiar los permisos/dueño/grupo de un fichero/directorio?:

Para cambiar el dueño del fichero: `chown` *usuario fichero*
Para cambiar el grupo del fichero: `chgrp` *usuario fichero*
Para cambiar los permisos se utiliza el comando: `chmod` *permisos fichero*

Los permisos se pueden especificar de diferentes maneras, una serie de ejemplos, es lo mejor para comprenderlo:

```
chmod ugo+rwx test (da permisos rwx a todos, user,group,others)
chmod ugo-x test (quita permiso x (ejecucion) a todos, user,group,others)
chmod o-rwx test (quita permisos rwx a others)
chmod u=rwx,g=rx test (da permisos rwx a user, rx a group y ninguno a others)
```

Asi podrias continuar con todas las posibles combinaciones de letras, es cuestion de usar la imaginacion ;-)

Existe otro metodo que utiliza numeros, en vez de letras para asignar permisos, la siguiente tabla nos puede ayudar un poco a comprender esta manera:

```
r w x  VALOR DECIMAL

0 0 0  0 (000 binario es 0 en decimal)
0 0 1  1  .........
0 1 0  2  .........
0 1 1  3  .........
1 0 0  4 (100 binario es 4 en decimal)
1 0 1  5  .........
1 1 0  6  .........
1 1 1  7 (111 binario es 7 en decimal)
```

1 significa activado y 0 desactivado, o sea 101 activa `r` y `x`, y desactiva `w`.

Sabiendo esto solo tenemos que usar el valor decimal, un ejemplo aclarara esto.

```
chmod 750 test
```

```
da permisos rwx al usuario (7=111)
da permisos r-x al grupo (5=101)
da permisos --- a los demas (0=000)
```

7.12 ¿Cómo apago mi equipo?

Para apagar el equipo tienes que hacerte superusuario y hacer:

```
[asimovI@localdomain root]#  shutdown -t3 -h now
```

Esto pararía todos los procesos en curso, haría `sync`, entraría en el runlevel 0, y desmontaría todas las particiones. El proceso de parada llegaría cuando se observa `system halted`, y si la fuente lo permite y se compiló el kernel con esa opción, el ordenador se apaga automaticamente sin necesidad de pulsar el boton de apagado.

La opción h viene de `halt` (parar), y hace que se mantenga parado después de todo el proceso de apagado. La opción `now` indica que lo haga ahora, esta opción se puede sustituir por un número los minutos a esperar antes de realizar el `shutdown`.

Para reiniciar el equipo tienes que hacerte superusuario y hacer:

```
[asimovI@localdomain root]#  shutdown -t3 -r now
```

Pues como habías imaginado hace lo mismo que la anterior pero en vez de ponerse en `system halted` se pone en `system reboot` y reinicia.

Existe otro método que consiste el accionar tres teclas, sí, esas que tanto hay que usar en otros sistemas operativos (aunque cada vez menos ya que apenas los usamos), las CAS (Ctrl+Alt+Supr). Pues bien, en el proceso de arranque de Linux, cuando lee la `/etc/inittab` existe una entrada que asocia estas tres teclas a un comando. Si lo ves verás que lo asocia al comando anterior.

```
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Un consejo:

En los métodos anteriores es necesario ser superusuario ya que el comando `shutdown` lo requiere. Cosa lógica ya que sino cualquier usuario de la máquina lo podría echar abajo. Esto hace bastante incómodo el apagarlo, por lo que un consejo práctico es ir a la `/etc/inittab` y cambiar la `-r` por la `-h` del comando que se une a las CAS, esto asociará el apagar el sistema a las tres teclas y no al reiniciarlo. Con esto ya no es necesario hacerse superusuario, sino estar sentado delante del ordenador a apagar.

Nota: En los kerneles 2.2.x hay que modificar una línea en el `/etc/rc.d/*halt`. En dónde llame al halt, hay que añadirle la opción -p. Si no lo que hace es reboot.

8 Internet / Redes / Modems

8.1 ¿Qué es un modem? ¿Y un puerto serie?

¿Por qué los modems están normalmente asociados a puertos serie? Lo impone el hecho que cuando uno usa un modem, la línea telefónica a la que está conectado sólo tiene un cable por el que enviar la información. Para transmitir simultáneamente los 8 bits de los que consta un byte, que es la unidad de información usual en los ordenadores, harían falta 8 cables (8 líneas telefonicas). Una solución alternativa es enviar los 8 bits sucesivamente uno detrás del otro a intervalos de tiempo regulares. Precisamente esto es lo que hace un puerto serie.

El corazón del puerto serie es un chip del ordenador llamado UART (Receptor-Transmisor Asíncrono Universal) el cual se encarga de todo el trabajo, tanto para recibir datos como para enviarlos. Así, la potencia de dicha UART y lo bien configurada que esté influirán en la calidad de la comunicación.

El flujo de bits generado por la UART/puerto serie no puede introducirse directamente en la línea telefónica al no estar preparada para ello. La solución es utilizar un aparato intermedio que tome los bits que llegan del puerto serie y emita un tono u otro por la línea telefónica según le llegue un bit 0 o un 1. Esto es lo que se conoce como "modular una señal". Al otro lado del hilo telefónico otro aparato similar interpretaría estos tonos y generaría unos o ceros en función del tono recibido. Éste sería el paso opuesto, "demodular la señal". Como normalmente se deseará enviar y recibir, el aparato intermedio deberá realizar ambas funciones: MOdular y DEModular (MO-DEM). Así funcionaban los modems antiguos. Los modernos se basan en los mismos principios aunque son infinitamente más versátiles y potentes.

Los modems modernos realizan la modulación y demodulación utilizando chips especiales llamados DSP (Procesadores Digitales de Señales en castellano). Son como microprocesadores, pero muy rápidos y optimizados para tratamiento de señales. Los modems modernos también incorporan sistemas de compresión de datos que aceleran las transferencias, sistemas de corrección de errores, cambios automáticos de velocidad de transferencia si la calidad de la señal mejora o empeora, etc...

Existen dos clases de modems: Los internos y los externos. Los externos van conectados a un puerto serie del ordenador por lo que simplemente se limitan a las funciones descritas de modulación-demodulación, compresión, correción, etc. Los modems internos "de verdad", al no ir conectados a un puerto serie incorporan una UART como las de dichos puertos serie, de forma que a ojos del ordenador se trata de un puerto serie más.

8.2 Entonces, ¿Qué es un winmodem, modem HSP o modem software ? ¿Funcionan los winmodem en Linux ? ¿Y los modems PCI?

Son modems internos "capados". Los fabricantes les quitan chips para que sean más baratos. Por ejemplo, algunos no tienen UART, a otros les quitan los protocolos de compresión, etc. Para que tengan las mismas funciones que los modems "de verdad", las funciones correspondientes a los chips retirados las tienen que realizar drivers del sistema operativo, o sea el microprocesador del ordenador.

Los fabricantes de los distintos modems sólo ofrecen drivers para Windows y no dan especificaciones de como se programan, y así desarrollar versiones para Linux. Por tanto estos modems no funcionan en Linux.

Sobre los modems PCI, no tengo noticias de que ninguno sea un modem "de verdad". Es probable que en el futuro aparezcan pues el hecho de ser PCI no es ningún impedimento para ello.

8.3 ¿Cómo se configura el puerto serie ?

La primera pregunta sería: ¿ Debo configurar mis puertos serie ?

En general, sólo deberás configurar los puertos serie en estos casos:

- Tu modem es interno y Plug'n'Play. Deberás configurar el modem y al menos el puerto serie que incorpora.
- Tienes más de dos puertos serie, contando además con los que incorporan los modems internos (si tienes alguno). Por lo general deberás configurar todos los puertos a partir del COM3 incluido.

- Un tercer caso, más raro, que consistiría en tener un modem interno en COM1 o COM2 y que no usara los recursos estandar (puerto 3f8 e interrupcion 4 para COM1, y puerto 2f8 e interrupción 3 para COM2). Los recursos utilizados por el modem pueden consultarse en el Panel de Control de Windows 95 (Sistema -> Administrador de dispositivos -> Puertos COM y LPT -> el puerto que sea -> Recursos). Entonces deberás configurar el puerto del modem.

Si no cumples ninguna de estas condiciones, puedes ignorar el resto de la sección.

Un puerto serie (más exactamente, la UART del puerto serie) usa un rango de direcciones de entrada-salida y una solicitud de interrupción para comunicarse con el sistema operativo. Para el correcto funcionamiento del ordenador y del sistema operativo, ningún otro dispositivo del ordenador puede usar alguno de esos valores. Nuestro objetivo será conseguir esto, y en segundo lugar, hacer que el sistema operativo atienda a ese rango de entrada-salida y esa interrupción para comunicarse con el modem.

Tradicionalmente la elección del rango de entrada-salida y la interrupción se realizaba a través de una serie de microinterruptores manuales situados en el modem. Conocida la posición de estos interruptores, se configuraba el sistema operativo para que usara esos valores. Hoy en día casi la totalidad de los modems internos son Plug'n'Play, lo cual significa que el sistema operativo puede, por software, decirle a la tarjeta qué valores debe usar.

Configuración de modems Plug'n'Play en Linux

Si tienes un modem Plug'n'Play, lo primero que habrá que hacer es decirle que rango de entrada-salida y que interrupción debe usar. Lo más cómodo en este caso es usar los mismos valores que Windows 95. Para ello arranca Windows 95, y haz Inicio->Configuración->Panel de control. Una vez en el Panel de control, elige el icono 'Sistema' y dentro de él, el 'Administrador de dispositivos'. Dentro del apartado 'Puertos COM y LPT' elige el correspondiente al puerto COM donde tienes el modem. Busca en 'Recursos' el rango de entrada-salida y la interrupción que use y apúntalos.

Con esos valores en la mano, puedes configurar tu modem Plug'n'Play en Linux. Eso se hace con el programa 'isapnp'. Entra en Linux y busca si tienes un archivo llamado `/etc/isapnp.conf`. Si no lo tienes haz lo siguiente:

```
pnpdump > /etc/isapnp.conf
```

Esto generará el esqueleto de un archivo de configuración para 'isapnp'. Este programa generará un listado con todas las posibles posibilidades de configuración que permiten tus dispositivos Plug'n'Play, pero todas deshabilitadas con un '#' al principio de cada línea. Los dispositivos aparecen por secciones por lo que es fácil encontrar los correspondientes a tu modem.

Así tu misión se reducirá a encontrar en la sección correspondiente a tu modem del archivo `/etc/isapnp.conf`, dos líneas así:

```
# (IO 0 (BASE 0x0NNN))
# (INT 0 (IRQ MM (MODE +E)))
```

donde NNN sea el rango de entrada-salida que use el modem en Windows y MM la interrupción. Las líneas no tienen por qué ser consecutivas ni ir en ningún orden especial. Cuando las encuentres, quítales el '#' inicial. Luego, al final de la sección del modem encontrarás una línea parecida a esta:

```
# (ACT Y)
```

De la misma forma, le quitas el '#' inicial. Ya está. Listo para enviar dichos valores al modem. Esto lo hace la orden `isapnp /etc/isapnp.conf` y deberá aparecer en los scripts de arranque de Linux para enviar los valores cada vez que enciendas el ordenador. Las distribuciones más recientes (Red-Hat 5.0 y posteriores, Debian 2.0 y posteriores) ya incluyen dicha invocación por lo que no necesitarás tocar los scripts de arranque. En caso de tener una distribución antigua, será necesario añadir dicha orden. Un buen sitio para incluirla sería el script `rc.local`, que suele estar situado en uno de los subdirectorios de `/etc`.

De todas formas será necesario ejecutar `isapnp /etc/isapnp.conf` si queremos activar los cambios en este momento y nos queremos ahorrar el rearrancar la máquina.

Ahora hay que decirle al sistema operativo que debe usar esos valores para comunicarse con el modem. En el caso de los modems Plug'n'Play lo que viene a continuación es imprescindible.

Configuración de los puertos serie a ojos del sistema operativo

Lo primero que hay que saber es que Linux no llama a los puertos serie de la misma forma que el DOS o Windows. Así existe la siguiente correspondencia:

```
DOS-Windows                Linux
-----------              ----------
   COM1                  /dev/ttyS0
   COM2                  /dev/ttyS1
   COM3                  /dev/ttyS2
   COM4                  /dev/ttyS3
```

Ojo a las mayúsculas y las minúsculas del nombre de Linux. Son importantes.

Para leer la configuración y configurar los puertos serie se usa la orden 'setserial'. Para leer la configuración, hay que ejecutar `setserial` con el puerto serie que deseamos ver como único parámetro.

Por ejemplo:

```
setserial /dev/ttyS1
```

nos muestra la configuración de COM2 y se obtendrá algo como esto:

```
/dev/ttyS1, UART: 16550A, Port: 0x02f8, IRQ 3
```

o sea, el puerto serie (`/dev/ttyS1`), el tipo de chip UART que lleva (16550A), el puerto base de entrada-salida (2f8) y la solicitud de interrupción (3). Nótese que estos son los valores que el sistema operativo CREE que debe utilizar para comunicarse con el modem/puerto serie. Es posible que ahora no coincidan con los reales, incluso que ni siquiera detecte la UART, lo cual no es raro en caso de modems Plug'n'Play.

Si no coinciden, asignaremos los valores adecuados. Por fortuna, por lo general no será necesario indicar la dirección base de entrada-salida. Salvo casos muy, muy excepcionales COM1 usa la dirección base 3f8, COM2 la 2f8, COM3 la 3e8 y COM4 la 2e8, y eso setserial lo sabe. Así, al indicar el puerto serie, setserial implícitamente sabe cual es la dirección base. En cambio, en caso de tener un puerto COM5 o superior sí será necesario indicar la dirección base.

Así para COM1-COM4 sólo indicaremos la interrupción. Por ejemplo, supongamos que tenemos un modem en COM3 que usa la interrupcion 5. La orden a introducir sería.

```
setserial /dev/ttyS2 irq 5 autoconfig
```

Y si quisiéramos indicar también la dirección base de entrada-salida (por tener el modem en COM5, por ejemplo) se haría así.

```
setserial /dev/ttyS4 port 0x2e8 irq 10 autoconfig
```

lo cual hace que el sistema operativo use un rango de direcciones desde la 2e8, y la interrupción 10. Nótese que el '0x' antes de la dirección base es importante.

Por último sólo queda ver que los cambios han surtido efecto. Para ello bastará usar de nuevo setserial de la primera forma que se indicó.

Por desgracia esta configuración durará hasta que apaguemos el ordenador. Por tanto será necesario incluir la línea o líneas de configuración `setserial /dev/ttyS2 irq 5...` en uno de los scripts de arranque de Linux para que los puertos se configuren cada vez que se inicie el sistema operativo. Un buen candidato es `rc.local`, que estará en uno de los subdirectorios de `/etc`. En caso de ser necesario incluir en `rc.local` tanto el `isapnp /etc/isapnp.conf` para configurar un modem Plug'n'Play, y el `setserial...` de los puertos serie, deberá colocarse el `isapnp...` en primer lugar.

También habrá que usar `setserial` de la misma forma si el ordenador tuviera más de dos puertos serie, (contando los que traen los modems internos en caso de disponer de alguno). El motivo es que por alguna extraña razón de diseño del PC, COM1 y COM3 por defecto comparten la misma interrupción, y a pesar de que esa configuración no es deseable, Linux por defecto busca COM3 en la misma interrupción que COM1, por lo que habría que cambiar la de COM3. Lo mismo ocurre con COM2 y COM4 por lo que habría que cambiar COM4. Si no se hiciera esto, podrían ocurrir cosas muy raras, como ratones y modems que dejan de funcionar, modems que funcionan ª pedales"(cuando se mueve el ratón), etc...

8.4 ¿ Cómo compruebo que todo está bien configurado ?

Una vez activados los cambios con el comando isapnp para los modems Plug'n'Play y el/los comando(s) setserial necesarios para configurar el/los puerto(s) serie, o se ha reiniciado el ordenador con los scripts de arranque modificados para que ejecuten dichas órdenes (es más seguro hacer lo primero), se puede probar si todo va bien. La forma más sencilla es usar un programa de comunicaciones y enviarle comandos al modem para ver si responde.

Así un programa adecuado sería `minicom` que es un terminal de comunicaciones tipo Telix. No es complicado de configurar. Pulsando `Ctrl+a` y luego z sale un menú. Una de las opciones 'cOnfigure minicom' dará paso a otro menú donde la opción "Serial port setup"nos permitirá elegir el puerto serie, la velocidad con la que nos comunicaremos con él (no la velocidad del modem) y otras opciones más. Lo normal es usar 115200 o 57600 para la velocidad, 8N1, "Hardware Flow Control: Yes"y "Software Flow control: No". Una vez configurado y sin ningún menú sobre la pantalla escribes 'AT' (Las dos en mayúsculas o las dos en minúsculas) y pulsas ENTER. Si el modem responde al instante 'OK' o con un número cero, indicaría que el modem funciona y que responde bien. Para completar la faena puedes probar a conectarte a una BBS o a tu nodo de infovia plus (los nodos de infovia plus permiten una conexión tipo terminal, aunque no sé que se puede hacer con ella) y así probar que todo va bien.

Si el modem tarda en responder al 'AT', o no responde, algo ha fallado en la configuración del modem o del puerto serie y habría que revisar lo hecho.

Un comentario sobre el 'AT'. Cuando se escriben estas dos letras pueden ocurrir varias cosas.

- Que salga el AT mientras lo escribes (lo normal).
- Que no te salga el AT, pero al pulsar `ENTER` el modem responda con OK. Para solucionar esto deberías activar el eco local del terminal con `Ctrl+a e` o escribir a ciegas "ATE1"y pulsar `ENTER`.
- Que te salga AATT y todo lo que pongas por duplicado. No es problema porque el modem sólo recibirá 'AT'. Para quitar esto, desactiva el eco local del terminal con `Ctrl+a e` o escribe "ATE0ªunque te salga duplicado. Cuando pulses ENTER tras el ATE0 ya funcionará bien.

8.5 ¿Cómo conecto a internet con pppd ?

¿Que es pppd? Se trata de la primera posibilidad de conexión a internet que ofrece Linux. Es una conexión a internet que gasta pocos recursos y con posibilidad de hacerla desde la consola, sin necesidad de entrar en modo gráfico. Cualquiera que aspire a obtener al máximo rendimiento de la conexión a internet, por ejemplo automatizando la conexión, automatizando la recogida de correo, noticias, ftp, etc... casi seguro no tendrá más remedio que pasar por aquí.

Como inconvenientes, tiene unos cuantos. El más destacable es la falta total y absoluta de interactividad o de información en pantalla de cómo va la conexión, siendo necesario el uso de otros comandos (como `ifconfig`) o la observación de archivos históricos (`/var/log/messages`) para averiguar si la conexión se ha realizado con éxito o ha habido algún error.

La configuración de pppd se basa en la creación de una serie de archivos.

`/etc/ppp/options` : Configuración de la conexión y del puerto serie
`/etc/ppp/marcado` : Configuración del modem y marcado del número de teléfono
`/etc/ppp/pap-secrets` : Claves de acceso para identificación con PAP
`/etc/ppp/chap-secrets` : Claves de acceso para identificación con CHAP
`/etc/resolv.conf` : Servidores de nombres (DNS)

Así un posible `/etc/ppp/options` sería así:

```
-----------------------------------------------------------------
connect "/usr/sbin/chat -v -f /etc/ppp/marcado"
name mi_login@mi_proveedor
defaultroute
noipdefault
modem
crstcts
asyncmap a0000
mru 576
/dev/ttySx
115200
--------------------------------------------------------------------
```

Comentarios:

- En mi_login@mi_proveedor irá el login y el proveedor (el proveedor sin dominio, por ejemplo: fulanito@arrakis)
- Donde pone `/dev/ttySx`, la 'x' indica el puerto serie: `/dev/ttyS0 = COM1, /dev/ttyS1 = COM2`, etc...
- La velocidad 115200 *NO es la velocidad del modem*, es la velocidad entre el modem y el puerto serie, que no tiene por qué coincidir con la velocidad en el lado de la línea telefónica. *La velocidad del modem (33600, 28800...) NO hay que ponerla en ningun sitio.* Algunos modems no soportan esa velocidad entre el modem y el puerto serie y habrá que probar con 57600 o incluso con 38400. Si se tiene una UART 16450 o 8250, utilizar 38400 (usar `setserial /dev/ttySx` para conocer la UART del modem o del puerto serie, donde 'x' indica el puerto serie de la misma forma que antes). Nótese que el valor óptimo es 115200 y conforme menor sea dicho valor, menos rendimiento le sacaremos al modem.

`/etc/ppp/marcado`:

```
------------------------------------------------------------
ABORT "BUSY"
ABORT "NO CARRIER"
ABORT "NO DIALTONE"
ABORT "ERROR"
""
"AT& F" TIMEOUT 5 OK
"ATDTnumero_de_telefono" TIMEOUT 100 CONNECT
-------------------------------------------------------------
```

Comentarios:

- Los ABORT iniciales sirven para que, si el modem encuentra una situación anómala (no hay tono de llamada, comunicando...) y responde alguno de esos mensajes (BUSY, NO CARRIER...), el pppd pare y no pierda tiempo hasta fallar el TIMEOUT 100 que sería el que acabase la conexión. El problema es que no todos los modems responden de la misma forma (por ejemplo, el mío responde "NO DIAL TONE"en vez de "NO DIALTONE") y la correspondencia debe ser exacta. Por tanto, para una configuración correcta, será necesario consultar en el manual del modem cómo son estos mensajes.
- El "AT& F"es la cadena de inicialización del modem. La que se indica es de las más estandar. No obstante, en algunos modems puede ser mejor "ATZº variantes de "AT& F"(como "AT& F1", "AT& F2"...). Como referencia, se debe usar aquella que configure el modem para usar control de flujo hardware (RTS/CTS). En caso de no acertar en la configuración del modem, es posible que se produzcan algunos efectos indeseables (el modem no cuelga al cortar la comunicación, conexión lenta...). Una posibilidad es "fusilar"la que use windows. Se puede ver dentro del directorio de windows un archivo llamado modemlog.txt o modemdet.txt.
- El numero_de_telefono es precisamente eso. El número de telefono del nodo de Infovia Plus, el de Iddeo, etc...

`/etc/ppp/pap-secrets` y `/etc/ppp/chap-secrets`

Existen dos formas posibles de identificación cuando nos conectamos a un proveedor: PAP y CHAP. PAP es la más sencilla y es la que usaba por ejemplo Infovia. CHAP es muy usada en Infovia Plus.

El formato de ambos archivos es el mismo:

```
------------------------------------------------------------------------
mi_login@mi_proveedor *       password
------------------------------------------------------------------------
```

Comentarios:

- mi_login@mi_proveedor debe ser exactamente lo mismo que se puso como parámetro de la opción 'name' en `/etc/ppp/options`.
- Los campos *Van separador por TABULADORES no por espacios*.
- Si se dispone de más cuentas en otros proveedores, se pueden añadir nuevas líneas con el mismo formato.

¿Cual uso, pap-secrets o chap-secrets? Si sabes que tu proveedor usa PAP, grábalo como `pap-secrets`. Si usa CHAP, grábalo como `chap-secrets`. Si no sabes cual usa, puedes hacer prueba-y-error o crearte los dos, o crearte uno sólo y hacer un enlace simbólico de uno al otro. Por ejemplo, si has creado chap-secrets, para hacer un enlace desde pap-secrets se haría:

```
cd /etc/ppp
ln -s chap-secrets pap-secrets
```

Esto en teoría haría que pppd y tu proveedor se pusieran de acuerdo para elegir PAP o CHAP según prefieran.

`/etc/resolv.conf` :

Aquí irían las direcciones de los DNS primario y secundario. El formato es muy sencillo:

```
-------------------------------------------------------
nameserver DNS_primario
nameserver DNS_secundario
-------------------------------------------------------
```

Por ejemplo, `nameserver 195.5.65.2`

Comentarios:

- En lugar de DNS_primario y DNS_secundario, se pondrán las direcciones numéricas de los DNS primario y secundario respectivamente. Los DNS primarios y secundarios deberás preguntarlos a tu proveedor y deberán dártelos, pues no son secretos.
- Si no tuvieras DNS_secundario, puedes ahorrarte la segunda línea.

Una vez creados los scripts, la conexión se efectuaría al escribir `pppd` o `/usr/sbin/pppd`.

Para comprobar si la conexión tiene éxito, se puede usar el comando `/sbin/ifconfig` y ver si al cabo de un minuto o dos aparece un bloque nuevo `ppp0`.

Otra forma es consultando periódicamente el fichero `/var/log/messages` hasta ver si aparece algo como esto:

```
Local  IP address xxx.xxx.xxx.xxx
Remote IP address yyy.yyy.yyy.yyy
```

Lo cual indicaría una conexión con éxito. Una forma de hacer esto sería con la orden `tail -f /var/log/messages` y pulsar `Ctrl+C` cuando veamos que la conexión ha funcionado o fallado.

La desconexión se realizaría introduciendo la orden `killall pppd`.

En caso de necesitar ayuda, remitir la duda a es.comp.os.linux.misc o es.comp.os.linux.redes mandando copia de los scripts (ocultando los passwords, claro) y lo que haya salido en `/var/log/messages` durante la conexión fallida.

NOTA IMPORTANTE: Los ficheros arriba descritos han de crearse desde cero con un editor de Linux. Crearlos con un editor de DOS o de Windows, o con operaciones de cortar-y-pegar en Windows añade un caracter no visible al final de cada línea que Linux confundiría con un caracter normal, inutilizando los scripts.

8.6 ¿Cómo conecto a internet con KDE-kppp?

Después de haber utilizado diversos sistemas para conectar con Internet a través de InfovíaPlus mediante módem y haber constatado en el grupo de noticias *es.comp.os.linux* la gran cantidad de veces que se pregunta acerca de cómo conectarse mediante kppp, hemos pensado que este breve resumen sobre como establecer la conexión usando el kppp podría ser interesante para algunos que se inician en linux y disfrutan del entorno grafico KDE. Las notas que escribo a continuación se refieren a la versión KDE 1.0 (las diferencias con respecto al KDE 1.1 son mínimas)y por lo tanto voy a partir de la base de que el usuario de linux ha instalado ya el KDE y le funciona sin problemas de ningún tipo.

Guía paso a paso de conexión a internet a través de InfovíaPlus con kppp.

Antes de comenzar

El módem debe estar correctamente instalado y su funcionamiento verificado (usando por ejemplo el programa minicom). El dispositivo asociado al módem dependerá del puerto serie al que está conectado. Si está conectado al "COM2"(en jerga DOS) el dispositivo será `/dev/ttyS1`, y podemos hacer un enlace simbólico a `/dev/modem`:

```
ln -s /dev/ttyS1 /dev/modem
```

Además debemos saber su velocidad máxima. Si el módem es de más de 28,8Kbps podemos tomar 115200.

Otros datos fundamentales son los que tienen que ver con nuestro proveedor de internet:

- *"Domain name"* (Nombre de dominio del servidor de acceso) seguido por un punto y la extensión del país: por ejemplo proveedor.es (se trata de un nombre inventado para ilustrar el procedimiento)
- *"DNS IP"* (Dirección IP del servidor de nombres de dominio: en nuestro caso será (inventado también) 194.150.24.25
- *Login:* compuesto por nuestro indicador de usuario (p.ej. user) y "Domain name"sin extensión separados por arroba: user@proveedor
- *Número de teléfono* que corresponda según nuestro nodo de conexión. Cuando existía Infovía este número era el 055, pero con InfovíaPlus, dependiendo del lugar hay un teléfono de acceso, por ejemplo (para Sevilla) 954547000

Comenzamos

Invoquemos kppp mediante acción sobre el correspondiente icono del menú del KDE o escribiendo en un terminal

```
[user@localhost]# kppp
```

Para que programa funcione para un usuario normal el propietario y el grupo deben ser root y el ejecutable debe tener el bit SETUID activado, para poder abrir el dispositivo del módem. Si no es así se pone uno superusuario

```
[user@localhost]# su root

(... da la password... y entonces hacemos)

[user@localhost]# chown root.root kppp
[user@localhost]# chmod +s kppp
[user@localhost]# exit

(...para volver a usuario normal...)
```

Si todo ha ido bien aparecerá la ventana del kppp y con el ratón accionaremos la tecla

```
setup
```

Entonces se abrirá la ventana kppp configuration con varias pestañas indicando secciones: Accounts; Device; Modem; PPP y About. Comenzaremos por la primera sección

- ***Seccion ACCOUNTS*** Accionamos la tecla new y aparece una nueva ventana llamada new account con otra serie de subsecciones: desde Dial hasta Accounting.

  Subseccion DIAL

  Esta sección contiene el Dial Setup que debemos rellenar:
  Connection name: Ponemos el nombre que queramos dar a la conexión; por ejemplo INFOPLUS
  Phone number: 954547000
  Authentication: La autenticación puede realizarse de cuatro modos; script-based, PAP, terminal-based o CHAT; debemos elegir la opción PAP
  Store password: Si no quieres tener que teclear la password cada vez que te conectes deja activada esta opción *Execute program upon connect*: Debe tener su casillero vacío. No escribas nada.
  Execute program upon disconnect: Debe tener su casillero vacío. No escribas nada.
  Edit pppd arguments: Una vez activada la opción los argumentos se introducen escribiéndolos en el casillero ªrguments"y pulsando la tecla ªdd". Deben estar los siguientes:

  ```
  detach
  defaultroute
  asyncmap a0000
  noipdefault
  ipcp-accept-local
  ipcp-accept-remote
  ```

  cuando hayamos terminado, pulsamos "close"y pasamos a las subsecciones siguientes

  NOTA: Estos argumentos los tenía así establecidos para cuando conectaba con Infovía. Al pasar a InfovíaPlus, alguno puede quedar superfluo, pero ¡funciona!, así que no quité ninguno.

  Subseccion IP

  Contiene el IP setup. La única opción que debe estar activada es *Dynamic IP address*

  Subseccion DNS

  Aquí tenemos el menú del DNS Server:
  Domain name: proveedor.es
  DNS IP address: La(s) introducimos en este casillero: 194.150.24.25 y pulsando la tecla ªdd"la(s) pasamos al DNS address list. Al menos debe haber una.
  Disable existing DNS servers during connection: Esta opción debe estar activada

  Subseccion GATEWAY

  Contiene el Gateway setup las opciones que deben activarse son:
  Default Gateway
  Assign the default route to this gateway

  Subseccion LOGIN SCRIPT

  ¡Principiantes Ni tocarlo! El edit script deberá estar vacío, con el primer casillero de opciones presentando la opción "EXPECT".

  Subseccion ACCOUNTING

  La opción *Enable accounting* debe estar activada. Selecciona entonces con la opción "Selected": Spain y dentro de ella, Telefónica Local y deja la opción "Volume accounting: No accounting"desactivada.

Entonces, una vez que hemos rellenado todas las subsecciones de la sección ACCOUNTS, accionamos el botón OK de la ventana "new account"y volvemos a la de kppp configuration. Hay que rellenar el resto de las secciones.

- ***Seccion DEVICE*** Aquí está el menú Serial device:
  Modem device: pon el dispositivo correspondiente, p.ej. /dev/ttyS1 (para modem en puerto serie com2)
  Flow control: CRTSCTS (hardware flow control)
  Line termination: La que corresponda. Generalmente es CR/LF
  Connection speed: La que corresponda, p.ej. 115200
  Modem lock file: /var/lock/LCK..modem

  Este fichero debe estar por defecto. No lo cambies si no sabes lo que haces, pues algunos programas como mgetty dependen de la existencia de tal fichero y kppp no funcionará.

- ***Seccion MODEM*** *Busy wait*: 0 seconds
  Modem Asserts CD line: opción desactivada
  Modem commands: Deja los valores por defecto y no cambies nada
  Query modem: Pulsando este botón kppp insta al módem a que se identifique. Sirve para probar el módem.
  Terminal: También sirve para comprobar si el módem negocia bien el protocolo de conexión. De todos modos no te detengas en esto si no sabes lo que haces y sigue adelante.

- ***Seccion PPP*** Presenta el ppp setup:
  pppd timeout: 60 seconds
  Dock into the panel on connect: opción desactivada
  Automatic redial on disconnect: opción generalmente desactivada
  Show clock on caption: opción activada
  Disconnect on X-server shutdown: opción activada
  Quit on disconnect: opción desactivada
  Minimize window on connect: opción activada

La sección ABOUT evidentemente es para mirar solamente. Por último accionamos el botón OK de la ventana kppp configuration. Con ello volvemos a la inicial de kppp. Activamos "Show log window". En el casillero "Connect to"deberá aparecer el nombre de nuestra conexión (en el ejemplo INFOPLUS). El login ID será en nuestro ejemplo user@proveedor. Escribimos la password en su casillero: ****** y por último pulsamos "connect"

Si todo va bien, aparecerán dos ventanas: una con el marco superior indicando "Connecting to: INFOPLUS"que mostrará Dialing 954547000 y otra mayor, a la derecha de la primera indicando en su marco superior "Login Script Debug Window"y presentando una serie de acciones:

```
ATZ
OK
ATD954547000

(Mientras está "Expecting Connect")
```

cuando se ha establecido la conexión, aparece en esa misma pantalla

```
CONNECT 115200
```

e indicará en su marco inferior "running pppd"mientras la otra ventana indica "Logging on the network". Cuando la ventana mayor presente en su marco inferior "Done"; la otra más pequeña, que en su marco superior exhibirá el nombre de la conexión (INFOPLUS), se minimizará y ya estamos conectados.

Si restauramos la ventana de conexión INFOPLUS en un momento dado nos mostrará la siguiente información:

```
Connected at 115200
Time connected ....
Session bill...PTAS
Total bill ....PTAS
```

y dos teclas "Details"(para ver los detalles de la conexión con nuestro proveedor) y "Disconnect"evidentemente para desconectarnos. Si la pulsamos, aparecerá de nuevo la ventana kppp y entonces, para salir, presionaremos la tecla "quit".

Preguntas habituales

- kppp muestra "**pppd has died unexpectlyº "data timeout has expired"(pppd ha muerto inesperadamente o ha pasado el tiempo de espera) Estos mensajes pueden aparecer alguna que otra vez, aisladamente. Si siempre se producen, entonces hay un problema. Hemos de comprobar que:**

  - pppd es realmente un binario y no un script
  - pppd tiene el bit setuid activado
  - el fichero `/etc/pppd/options` existe pero está VACÍO
  - que no has usado la opción "lock"como argumento del pppd en la configuración del kppp
  - no uses el enlace simbólico `/dev/modem`; en su lugar usa el dispositivo real que corresponda, p.ej., `/dev/ttyS1`
  - asegúrate que tienes los permisos necesarios

  en caso de seguir igual, invoca:

  ```
  [user@localhost]# tail /var/log/messages
  ```

  para considerar las causas de posibles fallos. Es muy conveniente leer el dossier sobre kppp que aparece pulsando la tecla help de la ventana de kppp.

- **kppp indica "sorry, can't open the modem"(perdón, no puedo abrir el modem) Esto significa que kppp no tiene permisos para abrir el dispositivo modem o que se ha seleccionado un dispositivo incorrecto en el cuadro del modem. Comprueba que el dispositivo sea correcto y luego dale el bit setuid a kppp.**

- **kppp muestra "I can't create a modem lock file"(no puedo crear el fichero de cerradura) O bien le das a kppp el bit setuid o das acceso al fichero /var/lock a los usuarios normales**

- **kppp sólo espera (Expecting OK) - ¿Has jugado con la configuración de CR/LF?**
  - El modem quizá necesita más tiempo para inicializarse. Ve al cuadro "Modem commands"y aumenta el "Init Delay Time".

- **La conexión funciona, pero no se pueden arrancar aplicaciones Puede ser que esté activada la opción "Auto configure hostname"y el servidor X tiene problemas para conectar con tu**

nuevo servidor. Si no hay más remedio que usar dicha opción, tendrás que proporcionar los permisos adecuados tecleando xhost + antes de realizar la conexión. Pero ¡MUCHO CUIDADO! porque cualquier otra persona tendrá acceso a tu servidor de X.

- **kppp funciona, pero Netscape indica que "There is not a DNS entry for this server"(El servidor no tiene una entrada DNS) Intenta hacer un ping al número IP de otro servidor. Si funciona entonces hay que comprobar si le diste al kppp al menos una dirección DNS o si en el fichero `/etc/host.conf` hay una línea indicando algo parecido a `order hosts, bind`. (Debe haberla para que se consulte al servidor de nombres cuando se busca una dirección)**

8.7 Mi modem cuelga durante la negociación inicial.

La negociación se puede distinguir por ser la fase en la que suenan los modems. Si cuelga bruscamente puede ser debido a un TIMEOUT demasiado corto en el script de marcado. Se puede subir a TIMEOUT 100 o a TIMEOUT 150. En caso de seguir colgando, se puede intentar añadir `S7=100` o `S7=150` entre el `AT` y el `DTnumero_de_telefono` en el script de marcado.

8.8 Mi modem suena muy fuerte ¿Cómo le bajo el volumen?

Depende de lo que quieras. Si simplemente quieres bajar el volumen del modem, basta añadir L1 o L0 a la cadena de inicializacion del modem. Un posible sitio para colocarlo seria entre el "AT"y el `DTnumero_de_telefono` en el script de marcado.

Si lo que quieres es que no suene para nada (por ejemplo, si quieres automatizar la conexión para que se realice de madrugada) deberás añadir M0 a la cadena de inicialización del modem. Puedes colocarla en el mismo sitio que se indicó para L1 o L0.

8.9 Conecto a internet, pero no logro ir a ningún sitio.

Lo primero que hay que tener en cuenta es que no hay conexión hasta que no aparece por `/var/log/messages` lo de `Local IP Address xxx.xxx.xxx.xxx` y `Remote IP Address yyy.yyy.yyy.yyy` (o el bloque `ppp0` al escribir `ifconfig`. Esto viene al caso de que a veces la autentificación inicial falla, por ejemplo porque haya caído el proveedor o esté saturado. En esos casos Infovia Plus no corta la conexión y puede parecer que el modem está conectado a internet cuando en realidad ni siquiera ha llegado a contactar con el proveedor. En ese caso habría que revisar las claves de acceso que se hubiesen introducido en `pap-secrets` o `chap-secrets` o intentarlo más tarde.

En caso de aparecer `Local IP Address...` y `Remote IP Address...` en `/var/log/messages`, pero no conseguir conectar a ningún sitio habría que revisar en primer lugar `/etc/resolv.conf` y ver si las direcciones allí introducidas son correctas. Una forma de averiguar si es culpa de los DNS es probar a conectar usando una dirección numérica. Por ejemplo se puede probar con la dirección 130.206.1.2 que corresponde a www.rediris.es . Si conecta, es un problema en `/etc/resolv.conf` (o del DNS de tu proveedor).

Si se descarta el DNS, comprueba que tienes la opción *defaultroute* en `/etc/ppp/options`. En caso afirmativo, comprueba `/var/log/messages` y mira si aparece algún mensaje tipo `ppp not replacing existing default route to ....` Este último caso es típico si se tiene alguna tarjeta de red y has definido alguna dirección de gateway. Lo normal es eliminar dicho gateway. Averigua como introduciste el gateway y elimínalo. Una forma de eliminarlo sería por ejemplo con linuxconf, que viene con Red-Hat.

8.10 ¿Qué significa eso de Ünsupported protocol 0x31"?

Es un mensaje que aparece en `/var/log/messages` típico de Infovia Plus y debido a un protocolo de red llamado IPX. No debe suponer ningún problema para la conexión, pero si resultara molesto se podría añadir la opción *noipx* a `/etc/ppp/options`.

8.11 Mi modem no cuelga ¿Qué hago?

Asegúrate de que tienes (si usas pppd) la opcion *modem* en `/etc/ppp/options`.

En caso de que la tuvieras, comprueba que has puesto una cadena de inicialización al modem en el script de marcado (por ejemplo at& f, atz...).

En caso de que así siguiese sin colgar, tendrás que echar mano del manual del modem y buscar el comando "que haga que el modem cuelgue (o se resetee) cuando cambie la línea DTR". Normalmente es & Dx donde 'x' es un numero (que es lo que deberás buscar en el manual del modem, si prefieres prueba-y-error 'x' normalmente estará entre 0 y 4). Luego deberás editar el script de marcado del modem y colocar dicho "& Dx"entre el "AT"y el "DTnumero_de_telefono". Al usar el caracter '&', no olvides poner todo el conjunto (desde el AT hasta el final del número de teléfono) entre comillas dobles (p.e.: "AT& D3DT055").

8.12 Internet me va mucho mas lento con Linux que con Windows.

Esto puede deberse a muchas causas:

Deberás verificar la configuración del modem y del puerto serie. Sigue la descripción que se da en la sección correspondiente de la FAQ.

Comprueba que la velocidad que has puesto en `/etc/ppp/options` NO es la velocidad del modem (nada de 28800, 33600 o lo que sea, no es eso lo que debes poner ahí). Pon 115200 o 57600, preferiblemente la primera. Si tienes una UART 16450 u 8250 pon 38400 y asegúrate de NO poner el flag 'spd_hi' o 'spd_vhi' con setserial (si has seguido las instrucciones de la FAQ desde el principio no hace falta que compruebes esto). El tipo de UART que tienes lo podrás ver haciendo `setserial /dev/ttySx` donde 'x' es 0 para COM1, 1 para COM2, etc...

Si sigue lento o conectando mal, asegúrate de poner una cadena de inicialización al modem (ªt& f", ªtz") y que la opción *crtscts* aparece en `/etc/ppp/options`. En caso de continuar los problemas deberás echar mano del manual del modem y buscar las opciones que ªctiven el control de flujo RTS/CTS". Suelen ser de la familia de las que empiezan por '&'. En algunos modems es una sola orden (por ejemplo "& K3") y en otros hay que activar el RTS y el CTS separadamente. Colocar el o los comandos entre el "AT"y el "DTnumero_de_telefono". Si usas el caracter '&', asegúrate de poner todo el conjunto (desde el AT hasta el final del número de teléfono) entre comillas dobles (p.e. "AT& K3DT055").

8.13 Tengo más de un proveedor de internet ¿Qué hago para poder elegir por cual me conecto?

Una posible forma sería eliminar de `/etc/ppp/options` la opción *name mi_login@mi_proveedor*. De esta forma, si hemos puesto en `pap-secrets` o `chap-secrets`, el login y el password de todos nuestros proveedores, bastaría con escribir `pppd name login@proveedor` para elegir a cual nos queremos conectar.

Esto funciona si todos los proveedores comparten el mismo número de teléfono (por ejemplo, si todos están en infovía plus). En caso de tener diferentes números de teléfono, habría que eliminar en `connect "/usr/sbin/chat -v -f /etc/ppp/marcado"` de `/etc/ppp/options`, crearse un fichero de marcado particular para cada proveedor y crearse un fichero para cada proveedor en `/etc/ppp` con las dos opciones personalizadas que le correspondan. O sea:

```
/etc/ppp/nombre_del_proveedor:

------------------------------------------------------------
connect "/usr/sbin/chat -v -f /etc/ppp/marcado_proveedor"
name login@proveedor
------------------------------------------------------------
```

Y ejecutar pppd así:

```
/usr/sbin/pppd file /etc/ppp/nombre_del_proveedor
```

Las DNS de todos los proveedores se podrían poner en `/etc/resolv.conf`, añadiendo nuevas líneas "nameserver xxx.xxx.xxx.xxx". Es conveniente ordenar los nameservers poniendo primero todos los DNS primarios de todos los proveedores y después los DNS secundarios. También conviene poner en primer lugar los correspondientes a los proveedores más usados.

8.14 Solo puedo usar internet como root. ¿Qué hago para poder hacerlo como usuario?

Hay dos formas de hacerlo. La forma "buena", sería usando el programa `sudo` el cual permite ejecutar como usuario algunos programas que sólo podrían funcionar como root. Esto supondría que cada vez que uno quisiera conectarse a internet debería introducir un password, lo cual es útil en caso de que otros tuvieran acceso a nuestro ordenador. Para lo configuración de sudo consultar `man sudo`.

La forma "mala", y digo mala porque es más insegura en caso de que otros tengan acceso a nuestra máquina, consistiría en poner pppd "suid root"lo cual posibilita que un usuario normal tuviera acceso al programa de una forma similar que root.

Para esto último, habría que empezar por localizar `pppd`. Suele estar en el directorio `/usr/sbin`. Luego haría falta echar un vistazo para ver que permisos tiene. Esto se hace con la orden `ls -l pppd` y debe arrojar algo parecido a esto.

```
-rws--x--x   1 root   ppp     70336   May 18  1996 pppd
```

Lo que se persigue es que la tercera letra sea una 's', como en el ejemplo. Si el programa no está "suid root", la tercera letra será una 'x'. Entonces en este caso habría que hacer `chmod 4711 pppd` para conseguirlo.

Los scripts de `/etc/ppp` necesitarían tambien acceso libre para los usuarios así que habría que hacer `chmod 755 /etc/ppp` y `chmod a+r /etc/ppp/*`. El efecto secundario que esto tiene es que cualquier usuario del sistema puede leer las claves de acceso a nuestro proveedor de internet. Existe la posibilidad de limitar la conexión a internet (y a los scripts de conexión y claves de acceso), leer la seccion 7.11 (¿Cómo se cambian los permisos de ficheros y directorios?)

NOTA: Aún así, es posible que no seamos capaces de conectar con internet como usuario normal. Una posible causa de fallo sería intentar conectar a internet escribiendo `pppd` en vez de `/usr/sbin/pppd`. Para root funciona la primera, pero para el resto de usuarios no. El motivo es que el PATH (el conjunto de directorios en los que Linux busca un ejecutable) es diferente para root que para los usuarios normales. Los usuarios normales no tienen `/usr/sbin` dentro del PATH así que deberán hacer `/usr/sbin/pppd` para conectarse a internet o hacer un enlace simbólico a `/usr/sbin/pppd` desde otro directorio que sí esté en el PATH de usuario (como por ejemplo `/usr/bin`), o hacernos un script (pequeño programa) que se limite a ejecutar `/usr/sbin/pppd`. Para el enlace simbólico bastaría hacer: `ln -s /usr/sbin/pppd /usr/bin/pppd`

9 FTP/Telnet

9.1 ¿Qué es el protocolo FTP?

El Protocolo de Transferencia de Ficheros (File Transfer Protocol o, para abreviar, FTP) es uno de los pilares de Internet junto con el acceso a páginas Web (http). Durante mucho tiempo ha sido el servicio TCP/IP de Internet más utilizado, debido a que cualquier distribución Linux (así como en otros Sistemas Operativos) incluye el software necesario, muy fácil de usar y configurar, y de reducido tamaño.

Tal y como se comenta en la documentación (man ftp), "el Protocolo de Transferencia de Ficheros (FTP) es un protocolo de la familia TCP/IP usado para transferir ficheros entre máquinas que ejecutan TCP/IP (aunque hay programas estilo FTP para otros protocolos). Este protocolo permite transferir ficheros y manipular directorios en máquinas remotas. No está diseñado para permitir el acceso a otra máquina con el fin de ejecutar programas (para eso tenemos telnet), pero resulta la mejor utilidad para manipulación de ficheros.".

Esto quiere decir que una máquina con un servidor de ftp dispone de una estructura de directorios con ficheros y que además es capaz de permitir a cientos o miles de usuarios la recogida de ficheros o envío de nuevos ficheros al servidor.

Cuando hablamos de FTP estamos hablando (para explicarlo de una manera sencilla) de la descarga o subida de ficheros en Internet entre 2 máquinas, y es lo que permite entrar en ftp.idsoftware.com, por ejemplo, y bajarse la última versión del ejecutable de Quake II para Linux. También nos permite, por ejemplo, entrar en nuestro directorio privado en nuestro Proveedor de Servicios de Internet para poner allí nuestra página Web con el fin de que sea accesible desde el exterior. Para hacer esto disponemos de múltiples posibilidades debido a la disponibilidad de software para ello en Linux, desde ftp hasta ncftp, wget, etc..

Así pues, no es necesario utilizar el navegador de WWW para descargar programas de direcciones que comienzan por ftp://, ya que disponemos en nuestro sistema de herramientas mucho más potentes para tales menesteres: los programas de ftp. Aquí aprenderemos los comandos básicos así como la manera de instalar nuestro propio ftp anónimo para que otros accedan a nuestros ficheros desde el exterior.

9.2 El comando FTP

Comenzaremos hablando del programa ftp (lo que muchos habituales de Internet llaman «el típico ftp de toda la vida», conocido en las Universidades y sencillos terminales en modo texto cuando no se disponía (o no se necesitaba) del entorno gráfico.

Si se necesita transferir un fichero a otra máquina de una red o recuperarlo desde otra (y la red bien puede ser Internet), lo que necesitamos es un programa como ftp. Ftp es un programa sencillo y que viene en cualquier distribución de Linux (en el caso del autor, el paquete instalado es ftp-0.10-2.rpm). Como la mejor manera de aprender es con ejemplos (antes de ver todas las posibilidades que ofrece) supongamos que un compañero nos comenta que es posible obtener la última version de Netscape Communicator en el ftp de la dirección ftp://ftp.juan.es/pub/programas. Sería muy sencillo introducir esta URL (dirección de recurso de Internet) en Netscape Navigator para que éste nos haga de cliente de FTP, pero supongamos que no estamos dentro de XWindow (o que no nos apetece arrancar el navegador, o incluso que nos apetece bajarnos este fichero en varias sesiones, tal y como hacen GetRight y NetVampire para Windows, y que podremos hacer con ftp). Para conseguir nuestro objetivo hacemos:

```
[root@localhost sromero]# ftp ftp://ftp.juan.es
Connected to ftp.juan.es.
220 Juan FTP server (Version wu-2-41) ready.
Name (ftp.juan.es:sromero):_
```

En ese momento el programa habrá accedido a la máquina remota, quien nos está pidiendo que nos identifiquemos (Name) con nuestro nombre de usuario (también podríamos haber entrado con ftp y abrir el servidor con open ftp.juan.es). Si no somos usuarios habituales de dicha máquina (no poseemos ninguna cuenta especial en la misma), tendremos que identificarnos como anonimo (anonymous) y como password o palabra clave dar nuestra dirección de correo electrónico, proceso conocido como login o ftp anónimo, y que nos permitirá entrar en la máquina remota si ésta tiene permitido el acceso anónimo. Así pues contestamos:

```
Name (ftp.juan.es:sromero) anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:  santiago.romero@iname.com
```

Con lo que la máquina remota nos dará acceso a su sistema de ficheros ftp.

```
230- ¡Bienvenido al servidor FTP de Juanito!
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> _
```

En ese momento nos percatamos de que la terminal de texto se ha quedado en espera de introducción de comandos por nuestra parte. Ante la pregunta de «¿qué puede hacerse ahora?» surge la respuesta: «estamos en un sistema Unix», con lo que nada nos impide (si están dentro de aquel sistema) desplazarnos entre los distintos subdirectorios (ls, cd, etc.):

```
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 6
drwxr-xr-x   6 root     root    1024 Jul 17 10:47  .
drwxr-xr-x   6 root     root    1024 Jul 17 10:47  ..
drwxr-xr-x   2 root     root    1024 Jul 17 10:47  bin
drwxr-xr-x   2 root     root    1024 Jul 17 10:47  etc
drwxr-xr-x   2 root     root    1024 Jul 17 10:47  lib
drwxr-xr-x   2 root     ftp     1024 Jul 31 09:17  pub
226 Transfer complete.
```

A la vista de los directorios disponibles, vamos a pub/programas como se nos había especificado.

```
ftp> cd pub
250 CWD command successful.
ftp> cd programas
250 CWD command successful.
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 3
drwxr-xr-x   6 root     root         1024 Jul 17  09:18   .
drwxr-xr-x   6 root     root         1024 Jul 17  09:18   ..
drwxr-xr-x   6 root     root  10221024 Jul 17  09:18  netscape.rpm
```

Es posible hacer uso del comando get para descargar un fichero (o mget para múltiples ficheros), pero antes nótese que uno de los mensajes que se leyó al hacer el login fue Üsing binary mode to transfer files", lo cual nos indica que hay 2 maneras de bajarse ficheros: los de formato texto y los binarios, y que debemos especificar el modo correcto para bajarse ficheros de cada tipo (el comando ascii especifica formato ascii mientras que binary especifica formato binario). En este caso no tendremos que teclear binary pues el sistema ya nos avisa de que por defecto se descargan ficheros en binario, de modo que ejecutamos get para obtener el fichero:

```
ftp> get netscape.rpm
```

Con lo que obtendremos el fichero en nuestra máquina tras esperar el tiempo necesario y tras el cual nos avisará el propio programa de ftp.

9.3 Como bajarse ficheros en varias sesiones con FTP

¿y si cuando llevamos 1 hora de conexión necesitamos apagar el sistema o se corta la conexión? ¿Y si simplemente queremos bajarnos un fichero de gran tamaño en diferentes sesiones, conectándonos por ejemplo media hora diaria?

Nada nos impide continuar la descarga de un fichero otro día y desde el punto en que nos quedamos (sin perder ni un sólo bit) aunque se nos haya cortado la conexión en anteriores intentos. Para hacer esto disponemos del comando reget (con la misma sintaxis de uso que el comando get), el cual comprobará la existencia de un fichero con el mismo nombre que el que pretendemos bajarnos en el directorio actual de tal modo que en caso de existir continuará la descarga en el punto en que se quedo nuestro fichero incompleto, lo cual nos permitirá hacer de una manera muy cómoda y sencilla la descarga de grandes programas en varias sesiones, algo incluido en un programa con mucho tiempo y experiencia a sus espaldas y que algunos programas para Windows 95 pretenden vender como una gran novedad (véase GetRight o NetVampire).

Dicho de una manera sencilla: podemos comenzar a bajarnos un programa con get, cortar la descarga mediante CONTROL+C, salir del programa de ftp con quit, entrar al día siguiente y continuar con reget, repitiendo el proceso hasta completar la descarga o download del fichero.

```
ftp> get netscape.rpm
(al rato CTRL+C)

(Otro dia:)
ftp> reget netscape.rpm
```

9.4 Listado de comandos FTP

Otros comandos del ftp estándar son:

`!`: Ejecuta la shell para permitirnos la entrada de comandos en nuestro Linux. Con exit volveremos de nuevo al programa de ftp matando el shell. Si se especifica tras la admiración un comando de Linux, éste será ejecutado en la shell abierta.

Ejemplos:
!ls : haría un dir en nuestra máquina.
!xterm & lanzaría otra xterm
etc.

`ascii`: especifica tipo de transferencia de ficheros ASCII, en contraposición a ficheros binarios (no texto).

`binary`: especifica tipo de transferencia binaria (por defecto).

`bell`: le indica al sistema que ejecute un pitido (bell) cuando se finalicen la ejecución de los comandos. Así podemos ejecutar bell, y dejar un fichero de gran tamaño descargándose, sabiendo que tras su finalización oiremos un BEEP, lo cual nos permite dejar la ventana minimizada y hacer cualquier otra tarea.

`bye, quit`: termina la sesión ftp y sale.

`cd, dir, ls, list, mkdir, rmdir, pwd`: órdenes básicas de acceso a directorios, equivalentes a la del propio Sistema Operativo.

`close`: termina la sesión ftp sin salir del programa.

`delete y mdelete`: borran uno o varios ficheros en la máquina remota.

Ejemplos:
delete netscape.rpm
mdelete *.doc

`get y mget`: recibir uno o varios ficheros de la máquina remota.

Ejemplos:
mget *.doc
get "datos.doc"

`reget`: continuar bajando un fichero cortado anteriormente.

Ejemplo:
reget datos.doc

`put y mput`: enviar ficheros desde nuestra máquina a la máquina remota.

Ejemplo:
put fichero.html

`open`: conectar a un ftp remoto (ej: open ftp.xoom.com).

`rename`: renombrar fichero en la máquina remota.

Ejemplo: rename juan.html index.html

`user y pass`: especificar nuestro nuevo nombre y password.

`verbose`: activar o desactivar modalidad informativa.

`lcd`: Cambiar directorio local. Su utilidad radica en que cuando hacemos un download, éste irá a parar al directorio local donde estemos, de modo que cambiando de directorio podemos dejar los ficheros que nos bajamos en los lugares adecuados.

Ejemplo:
lcd /home/sromero/downloads

`help, ?`: imprimir información de ayuda del programa local.

Se puede obtener más información sobre estos comandos en la página man de ftp (man ftp), aunque la mayoría de los comandos de ftp son bastante explicativos y suelen coincidir con sus equivalentes en Linux.

9.5 Como subir paginas Web a nuestro ISP

Una de las cosas que pregunta mucha gente en los diferentes canales del IRC o en las news consiste en la subida de su página Web (creada en casa) a su directorio personal en su proveedor para que sea accesible al resto de usuarios de Internet. Aún hay mucha gente que no sabe realizar este proceso con lo que deja su espacio (que suele ser de 2Mb o más, dependiendo del servidor) libre en el servidor, y no aprovecha este servicio de Internet.

Lo primero de todo es crear nuestra página Web, que no es más que un directorio (con sus posibles subdirectorios) lleno de htmls, imágenes, y un fichero index.html que será la página inicial de la Web. Pero... ¿Cómo se pone dicha página en nuestro ISP?

Para solucionar esto disponemos de nuestro potente programa de ftp, que sirve también para este tipo de acciones. Para ello una vez creada nuestra página en casa, hacemos ftp a nuestro servidor (ejemplo: ftp ftp.arrakis.es), y el sistema nos pedirá la habitual identificación.

En lugar de identificarnos como anónimo (anonymous), lo hacemos con nuestro login y password (a veces necesitaremos utilizar los comandos user nombre, pass password) de acceso a nuestro servidor, con lo que tras la autentificación entramos al servidor dentro de nuestro directorio personal (si contratamos en su día el servicio de página Web), donde podremos preparar los directorios para nuestra página mediante los comandos mkdir y cd, que actúa sobre el servidor remoto, y lcd, que actúa cambiando de directorio en nuestra propia máquina.

Para ello hacemos `lcd /home/usuario/donde_sea/web` con lo cual ponemos el directorio actual en el especificado.

Una vez seleccionados lo directorios correctos, está disponible el comando put nombrefichero (o mput, para multiples ficheros) para subir la página (como otro fichero cualquiera) al servidor. Recuerde que la página principal de su Web debería llamarse index.html si quiere que haya alguna página por defecto en su dirección WWW, y que a partir de ésta debe estar la misma jerarquía de directorios que creó en su PC durante la creación de la misma.

Una vez subida la página, salga del programa de ftp con quit y haga un sencilla prueba mediante su navegador habitual.

9.6 Como instalar nuestro propio servidor FTP

Lo que se va a comentar a continuación es una manera muy sencilla de instalar un servidor de FTP anónimo para que otros usuarios puedan acceder a él mientras estemos conectados a Internet. Esto implica que podrán recoger ficheros así como dejarnos los ficheros que nosotros deseemos aceptar.

Para ello, y siguiendo la manera más sencilla posible, instalamos los paquetes anon-ftp y wu-ftpd y permitimos que estos habiliten la carga de los daemons necesarios en memoria (cosa que los paquetes rpm harán automáticamente al instalarlos con el parámetro -i):

```
[root@localhost sromero]# rpm -i anonftp-2.5-1.rpm
[root@localhost sromero]# rpm -i wu-ftpd-2.4.2b16-4.rpm
```

Si no disponemos de dichos paquetes los podemos obtener del ftp de RedHat (ftp.redhat.com), del subdirectorio pub/manhattan/contrib/i386, en formato rpm, para convertirlo a otros formatos de instalación (deb o tar.gz) mediante el script alien si fuera necesario. El primer paquete contiene la estructura básica de un servidor ftp mientras que el segundo contiene el servidor de ftp (un daemon) propiamente dicho.

El primero de los paquetes creará en /home/ftp una estructura de directorios que será el directorio raíz de nuestro ftp, siendo por defecto los siguientes:

```
[root@localhost home/ftp]# ls
bin    etc    lib    pub
```

Este directorio (/home/ftp) será el directorio raíz de los usuarios de nuestro ftp, es decir, que cuando el usuario realice un dir al directorio raíz estará viendo los mismos directorios que nosotros en /home/ftp. La forma de añadir o quitar programas a nuestro ftp resulta, pues, muy sencilla, y se reduce a añadir programas y directorios en el directorio /home/ftp/pub, que es de donde los usuarios remotos deben coger los ficheros.

En este subdirectorio crearemos los distintos directorios (ej: juegos, fotos, tutoriales, etc.) e incluiremos los ficheros que queremos que sean accesibles desde el exterior. Tras esto y un reset (o un simple reinicio del daemon) tendremos instalado nuestro servidor de ftp.

A la hora de decirle a alguien que puede entrar en nuestro servidor, recordemos que un usuario «de a pie» no suele disponer de una dirección de Internet propia (como ftp.netscape.com o ftp.idsoftware.com), sino que de lo único de que disponemos es (a menos que hayamos contratado nuestro propio dominio y dns) de una dirección IP a través de la cual somos accesibles desde el exterior. Si queremos que alguna persona del IRC (o, en general, cualquier usuario de Internet) entre en nuestro ftp, le deberemos comunicar nuestra IP ya que es perfectamente posible acceder a un ordenador remoto especificando la dirección IP en lugar del nombre. Para averiguar nuestra dirección IP basta con extraerla del archivo /var/log/messages, de los datos de la última conexión (al final del fichero):

```
fecha localhost pppd[393]: Remote message:
fecha localhost pppd[393]: local IP address 195.5.78.128
```

Si estamos dentro del IRC otra manera de obtener nuestra dirección IP es mediante comandos del IRC (como /whois, /dns, etc.) con nuestro nick identificativo (/whois Jose). Dicha IP se la comunicaremos a la persona que quiere entrar en nuestro ftp para que pueda hacerlo mediante la orden ftp dir_IP:

```
[pepe@OtraMaquina home]# ftp 195.5.78.128
```

Obviamente, parece muy incómodo tener que especificar cada vez nuestra nueva dirección IP, pero esto es así debido a que las direcciones IP asignadas por nuestro servidor varían en cada conexión, con las excepciones de que hayamos contratado una dirección IP fija (en cuyo caso no variará y todo el mundo podrá acceder a nuestro ftp usando la misma dirección) o de que hayamos contratado un nombre de dominio (como ftp.miftp.org), de manera que se pueda acceder al mismo desde esta notación nominal (mucho más sencillo). También es posible disponer de algún servicio de redirección de nombres a IPs, y hacer un ftp a dicho nombre.

Mediante el anterior comando, cualquier persona conectada a Internet (y mientras lo estemos nosotros), podrá acceder a nuestra jerarquía de directorios /home/ftp, y subir o descargar ficheros, así como utilizar todos los comandos de ftp que hemos visto en el presente artículo, y desde cualquiera de los clientes de ftp comentados (incluyendo, por ejemplo, Netscape) convirtiendo a nuestro ordenador en un servidor de ftp como otro cualquiera.

Si queremos testear si hemos configurado correctamente el ftp, podemos hacer un ftp sobre nuestra propia máquina de manera que se active nuestro servidor y podamos testear el servicio. Para ello especificaremos nuestra dirección IP de loopback (la dirección IP de nuestra máquina, 127.0.0.1), o el nombre de nuestra máquina (por ejemplo, localhost):

```
[root@localhost sromero]# ftp 127.0.0.1
[root@localhost sromero]# ftp localhost
```

En principio no debemos tener ningún problema en el test, aunque en caso de obtener un mensaje de «Connection refused», será necesario comprobar la existencia de la siguiente línea en el archivo de seguridad /etc/inetd.conf:

```
ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  in.ftpd -l -a
```

El objetivo de esto es permitir que nuestra máquina sea accesible vía FTP desde el exterior.

Existe un documento Howto sobre FTP anonimo que profundiza en la configuracion del mismo, haciendo especial incapie en los aspectos de seguridad.

9.7 Otros programas de FTP en modo texto

Pese a la gran versatilidad que tiene ftp, en Linux se disponen de algunos programas que pueden facilitar y acelerar algunas de las funciones de ftp, como ncftp (un cliente de ftp más avanzado que el sencillo ftp) y wget, (un programa que permite descargar ficheros de cualquier recurso WWW).

Ncftp se utiliza de forma similar a ftp, pero con la salvedad de que es capaz de hacer logins automáticos (user anonymous y password email), que permite edición de líneas anteriormente escritas (history de comandos), que dispone de un modo visual (a pantalla completa) y en línea (al estilo ftp), con una barra de estado que nos informará en todo momento de qué hace el programa, que recuerda URLs (si entramos una vez en hyper.unc.edu.es, bastaría en otra conexión especificar ºpen hyperº incluso ºpen hy"para que recuerde la URL completa), y con un comando get algo especial. El get que usa ncftp es capaz de saltarse los ficheros de los que ya disponemos, de coger múltiples ficheros (por ejemplo, get *), de bajar un programa a trozos sin necesidad de reget (mediante get -C nombre_fichero), bajarnos ficheros de X días de antigüedad o menos (get -n X *.txt), o incluso la posibilidad de hacer un get recursivo que entre en subdirectorios remotos y que restaure la misma jerarquía de directorios en nuestra máquina, así como los ficheros que contenga (get -R /pub/juegos). Además permite entrar directamente a ftps y directorios. Por ejemplo, en el programa de ftp básico, para entrar en ftp.test.com/pub/linux habría que hacer "ftp ftp.test.com"y luego "cd pub/linux", mientras que con ncftp podemos hacer directamente desde la línea de comandos:

```
ncftp ftp.test.com/pub/linux
```

Además ncftp recuerda el último directorio de cada site donde entremos, de modo que la siguiente vez que entremos en, por ejemplo, ftp.test.com apareceremos directamente en pub/linux. Dispone de muchas más funcionalidades, como un cálculo de tiempo esperado de download del fichero (ETA), una barra de progresión de downloads, etc. (para más información, consultar la página man).

Todo ello con un tamaño en disco de 174.000 bytes para la versión 2.4.3 (ncftp-2.4.3-3.i386.rpm), lo que lo hace ideal para bajarse cientos de imágenes de un directorio sin ir una por una, o subir multiples ficheros a una máquina remota (como cuando subimos nuestra página Web a nuestro ISP).

Wget es un programa que hace algo similar a get y reget pero desde la línea de comandos del Linux y que también sirve para direcciones http. La sintaxis de wget es la siguiente:

```
wget [opciones] dirección_de_Internet
```

Entre las opciones disponibles las más útiles son:

`-r`: wget recursivo (entra en directorios y links) `-L`: sólo entra en los links locales. `-c`: para bajarse un fichero en varias sesiones o continuar desde donde se quedó si se corta la conexión. Esta opción hace que si existe en el directorio actual un fichero con el mismo nombre que el que se pretende bajar, pero con menor tamaño, wget continue bajando datos sólo desde el final del fichero local (solo los datos que faltan para la finalización del download).

Un par de ejemplos de uso del programa podrían ser:

```
wget -c ftp.juan.es/pub/programas/netscape.rpm
wget -r ftp.juan.es/pub/fotos/paisajes
wget -rL www.8052.com
```

El primer ejemplo se bajaría el fichero netscape.rpm directamente al directorio actual, pudiendo cortarlo en cualquier momento con CTRL+C para continuar su download otro dia desde el punto en que se quedó (como reget, NetVampire, etc) utilizando el mismo comando. El segundo ejemplo se bajaría todos los ficheros que hubieran en el directorio pub/fotos/paisajes del ftp especificado, y el tercero se bajaría la web www.8052.com completa (incluyendo fotos, subdirectorios, etc) para poder navegar por ella en el disco duro offline.

Existen multitud de opciones (ver página del manual) que lo hacen indispensable y sumamente útil. Cuenta con un frontend para GTK en modo gráfico (Gwget) por si no nos gusta la línea de comandos, pero en sí wget es un programa ideal cuyo uso está justificado, pues podemos correrlo en diferentes xterm o consolas obteniendo ficheros de Internet a la máxima velocidad posible.

Otro programa bastante interesante y similar a Wget (pero que trabaja en forma de caché-proxy) es WW-Woffle, que permite descargar ficheros y páginas Web completas para navegar por ellas después en modo desconectado. También es posible marcarle páginas en modo desconectado para que se las baje cuando conectemos (sin ser necesaria mediación alguna por nuestra parte) de forma transparente al usuario.

Cftp (Comfortable FTP) es otro programa que nos permitirá navegar por el ftp mediante las teclas de los cursores, de forma que pulsando INTRO entraremos en los directorios sobre el que situemos el cursor virtual, o nos bajaremos el fichero sobre el que estemos.

Lftp es un programa al estilo ncftp con multitud de opciones de uso e ideal para la realización de mirrors (hacer una copia exacta de un ftp o de un subdirectorio del mismo) y mirrors inversos (copiar a un ftp el contenido exacto de un subdirectorio local, ideal para subir páginas Web completas con subdirectorios, ya que él mismo detecta qué ficheros locales son más nuevos que los presentes en el ordenador remoto, actualizando éstos). Como características fundamentales tiene: soporte de alias, caché, bookmarks, busqueda de ficheros, mirror (con opciones de continuación, recursión, borrado, etc.) mirror inverso, y sencillos ficheros de configuración.

9.8 Programas graficos de FTP

Si lo que queremos es que nuestro programa de ftp tenga un aspecto impecable y que sea una utilidad gráfica, disponemos en Linux de múltiples posibilidades (no sólo de texto vive el hombre, aunque siempre resulte lo más rápido y estable).

Entre los ftps más vistosos destacan gftp (del entorno gnome), IglooFTP, WX-ftp (clon de Wsftp) o el mismo Netscape Navigator.

10 Mail/News

10.1 ¿Qué es el correo electronico?

El correo electrónico es casi probablemente el servicio más útil para muchos de los usuarios de Internet, permitiendo intercambio de información casi instantáneamente. Este servicio consiste, dicho de una manera sencilla, en el envío de un bloque de información de texto (aunque puede llevar incluida información multimedia o en general cualquier otro fichero adjunto) de un remitente a un destino, de igual modo que el correo convencional. La diferencia radica en que en lugar de necesitar lapiz y papel, nuestro correo electronico (email, de electronic mail) es generado normalmente por nuestro programa de correo como un bloque de texto a enviar con el siguiente formato:

```
From:    <direccion_email_del_remitente>
To:      <direccion_destino_del_email>
Subject: <tema_del_que_se_habla>
Otros:   <otros_campos_de_informacion>
<texto_que_deseamos_enviar.>
.
```

Las direcciones de email son direcciones en servidores de correo igual que las direcciones postales lo son para la localización geográfica (por ejemplo, mi email es santiago.romero@iname.com, y un mensaje a dicha direccion llegaría a mi máquina servidora de correo de donde yo lo recogería para su lectura).

Un ejemplo de email seria el siguiente:

```
From: agonzalo@server.es
Date: Wed, 12 May 1999 22:19:24 +0200
Organization: Universidad de Sevilla
To: santiago.romero@iname.com
Subject: Una pregunta sobre configuraciones...

Hola, te queria hacer una pregunta...
¿me puedes ayudar con fetchmail?
```

Este bloque de texto se envia al servidor de correo de nuestro proveedor (o al nuestro, como ahora veremos) quien se encargara de moverlo por Internet hasta llegar al destinatario de nuestro mensaje. Podemos dividirlo en 2 partes: la cabecera del mensaje (todos los campos From, To, etc.), necesaria para el correcto envio del mismo, y el cuerpo del mensaje (el texto en si) que es lo que deseamos enviar.

Obviamente no tenemos que generar nosotros todo eso en un fichero de texto y despues realizar las operaciones para su envio (o recepción), sino que existen programas (clientes de correo) en los cuales introducimos el mensaje y éstos generan las cabeceras necesarias para los programas, permitiendonos además su envio y recepción. Ejemplos de clientes de correo son Pine, Netscape Communicator, MailX, Kmail, etc.

10.2 ¿Cómo puedo leer y contestar correo con mail/mailx?

El correo electrónico en Linux no puede considerarse como un extra al sistema operativo, sino que forma parte del mismo pues es una característica casi inherente a todo Linux. Esto quiere decir que aunque no hayamos instalado ningún cliente de correo como Netscape o Pine, probablemente dispondremos en el sistema de herramientas ya preparadas para ello.

Dentro de casi cualquier sistema Linux disponemos de un programa básico de email en consola (modo texto) llamado Mailx que nos permitirá el envio, lectura y contestación de mensajes de correo electrónico. En resumen, posibilita el envío de texto y datos dentro de una estructura coherente llamada mensaje, con su cabecera (campos: FROM (persona que envía el mensaje), TO (persona destinataria del mismo), SUBJECT (tema del que trata el mensaje) y CC o carbon copy (lista de personas que recibirán una copia del mismo).) y su BODY (cuerpo del mensaje), pudiendo incluir además ficheros añadidos que vienen incluidos dentro del paquete del mensaje (attach).

Para explicar el uso de mailx se va a usar un sencillo ejemplo en un máquina con un sólo usuario: sromero, además del obligatoriamente existente root (puede obtenerse la lista de usuarios presentes en el sistema y en qué terminal están trabajando simplemente ejecutando la orden who). Suponga os que somos el usuario sromero y queremos enviarle un mensaje a root para testear las capacidades de envio de email. La manera más sencilla de enviar un mensaje a un usuario de la red es ejecutar mail :

```
[sromero@localhost info]$ mail root
Subject: Prueba de MailX.
Esto es el cuerpo del mensaje. Es una prueba de mailx.
Un mensaje se acaba con un punto (.) al principio de línea.
.
Cc: (intro)
```

Al ejecutar mail root se nos ha preguntado el tema (subject), el cuerpo del mensaje (acabado en punto (.)), y a quién se le deseaba enviar copias del mismo (Cc), cosa que hubiera sido bastante útil si nuestro mensaje fuera una invitación a un grupo de amigos o una información para un grupo concreto de personas. Tras escribir esto, el mensaje escrito se graba como un fichero de texto (con el mismo nombre que el destinatario del mensaje, y al que se añadirán más mensajes conforme se le vayan enviando a dicho destinatario) en el directorio /var/spool/mail, a la espera de que el usuario destino los lea y vayan siendo eliminados del mismo.

Si root está conectado o entra más tarde en cualquier terminal del sistema, recibirá un mensaje de aviso indicando que el sistema ha detectado la llegada de correo para el usuario:

```
You have mail.
```

Para realizar la lectura del correo electrónico, simplemente se invoca al programa mail, obteniendo una lista del correo pendiente y esperando el programa a que el usuario lo lea, responda, borre o almacene para más tarde:

```
[root@localhost /root]# mail
Mail version 8.1 6/6/93.  Type ? for help.
"/var/spool/mail/root": 3 messages 3 new
>N  1 sromero   Thu Jul 30 17:04  17/55   "Prueba de MailX."
 N  2 sromero   Thu Jul 30 17:15  17/50   "Pregunta sobre adm"
 N  3 sromero   Thu Jul 30 17:16  15/44   "Felices vacaciones!"
& _
```

Como puede verse, mail nos muestra los mensajes pendientes (en este caso 3) por responder, con su usuario de origen (sromero), la fecha y el tema, con el cursor virtual (>) situado sobre el primero (es decir, indicando sobre cual actuará una orden de lectura, respuesta o borrado), y con un promtp (el carácter &) a la espera de recepción de órdenes sobre qué hacer con dicho correo.

Estas órdenes son:

```
'número' -> Listar el mensaje .
't' -> lista el mensaje actual (el apuntado por >)
'n' -> Pasar al siguiente mensaje y visualizarlo.
'd' -> Borrar mensaje.
'e' -> Editar mensaje
'r' -> Responder mensaje.
'q' -> Salir de mailx.
```

A la mayoría de estas opciones se les puede pasar un nº de mensaje (desde 1 hasta n) para indicar sobre qué mensaje realizar la acción (ejemplo 'd 3' o 'delete 3'). Otras opciones, obtenidas mediante la ayuda de mailx, son:

```
& ?
Mail Commands
t    listar mensaje
n    ir al mensaje especificado y listarlo.
e    editar mensaje
f    ver cabeceras del mensaje
d    borrar mensaje
s    añadir mensajes a un fichero
u    recuperar mensajes borrados
```

```
R    Responder a los remitentes del mensaje
r    Responder al remitente y a todos los destinatarios.
pre  hacer ir los mensajes de nuevo a /usr/spool/mail
m    enviar mensaje a los usuarios especificados.
q    salir grabando mensajes en mbox
h    mostrar cabeceras activas.
!    permite ejecutar una shell o comandos de shell
```

En el caso de no disponer de correo en ese momento, mail nos avisa con un:

```
 [root@localhost root]# mail
No mail for root
```

Otro fichero asociado es el fichero .signature de nuestro directorio home, utilizado como fichero de firmas y cuyo contenido es incluido al final de los emails escritos y respondidos por nosotros, pudiendo editarse y rellenarse con cualquier editor estándar de Linux para incluir nuestra firma personal. Este fichero es utilizado por la mayoría de gestores de correo de Linux como fichero de firmas por defecto, tal y como hace Mail, Pine, Netscape, Kmail, etc.

10.3 Envio de correo: Sendmail basico

Una de las necesidades básicas en nuestros sistemas Linux es el envio de correo electrónico o email a direcciones fuera de nuestra red local (a Internet, por ejemplo) o, en general, de nuestra propia máquina. Cuando escribimos un email en mailx, en Netscape, o en cualquier otro cliente de news, hace falta alguien que lleve dicho mensaje hasta el servidor de correo saliente (SMTP) de nuestro proveedor de Internet para que éste deje cada mensaje en el buzón destino correcto. Esta es una de las tareas que sendmail puede realizar en nuestras máquinas.

En nuestros envios de mensajes, pues, necesitamos un agente que extraiga dicho correo de `/var/spool` y lo envie a nuestro Proveedor de Servicios de Internet (por ejemplo, a smtp.arrakis.es). Sendmail es un programa tan completo que su estudio dispone de libros de cientos de paginas (incluso hay alguno que llega al millar de ellas) para abordar su gestión completa. Ya que muchos disponemos de Linux en nuestras propias máquinas (y somos también root de ellas), vamos a exponer el método más sencillo de configuración para uso con un sólo Proveedor de Servicios de Internet (ISP).

Lo primero es saber si lo tenemos instalado y con el daemon funcionando. Normalmente dicho paquete estará instalado y funcionado ya que es un daemon básico en todas las distribuciones Linux. Para arrancar el daemon (si no lo tenemos ya funcionando) y de paso chequear si está instalado, basta con ir a `/etc/rc.d/init.d` en el caso de Redhat (en otras distribuciones puede variar el directorio) y ejecutar:

```
/etc/rc.d/init.d/sendmail stop
/etc/rc.d/init.d/sendmail start
```

Obtendremos lo siguiente.

```
[root@compiler /root]# /etc/rc.d/init.d/sendmail stop
Shutting down sendmail: sendmail
[root@compiler /root]# /etc/rc.d/init.d/sendmail start
Starting sendmail: sendmail
```

Tambien podemos observar simplemente si durante el arranque y lanzamiento de los daemons del sistema sendmail es iniciado con lo que no haría falta su lanzamiento manual.

Tras asegurarnos de disponer de sendmail instalado y si somos root de nuestra máquina procederemos a la configuración más básica y sencilla del mismo (también es posible contactar con el administrador o root de la misma si no lo somos), que consiste en la edición del fichero de configuración `/etc/sendmail.cf`, (por ejemplo con el editor joe, emacs, jed, etc) con el fin de modificar algunas líneas del mismo y configurar sendmail para nuestro uso:

- Buscamos una línea similar a la siguiente:

  ```
  # "Smart" relay host (may be null)
  DS
  ```

  Esta línea hay que cambiarla de forma que referencie a nuestro servidor de correo saliente o smtp (para que sendmail envie a dicho servidor nuestro correo). A título de ejemplo, mi servidor de correo saliente es smtp.arrakis.es, por lo que la línea debe quedar como:

  ```
  # "Smart" relay host (may be null)
  DSsmtp.arrakis.es
  ```

  La dirección del servidor de correo saliente puede ser otra, pero suelen tener todas el mismo formato (stmp.ctv.es, smtp.arrakis.es, etc). De todas formas nuestro proveedor debe de facilitarnos el nombre de la máquina servidora de correo saliente al inscribirnos en dicho ISP.

- Ahora tenemos que enmascarar nuestra dirección de correo (es decir, modificar el fichero de configuración para que las direcciones de retorno de emails no sean las que tenemos en nuestra máquina (como sromero@localhost), sino las direcciones de retorno reales. Para ello buscamos la siguiente línea:

  ```
  # Who I masquerade as (null for no masquerading)
  DM
  ```

  A esta línea se le añade el dominio de nuestra dirección de correo. Si mi dirección email es sromero@arrakis.es, el dominio es arrakis.es:

  ```
  # Who I masquerade as (null for no masquerading)
  DMarrakis.es
  ```

Con estos cambios estamos forzando a que cuando se escribe un email, la dirección de retorno del mismo tenga el dominio correcto, cosa que se podría modificar de igual manera (sin el retoque de sendmail.cf) si nuestro programa para leer/contestar el correo permite hacerlo (en PINE, por ejemplo, modificamos la cabecera FROM en las opciones, en CUSTOMIZED-HEADERS). Para un usuario medio (acceso a Internet, recogida y envio de correo en un proveedor, etc.), el método comentado del sendmail.cf es más que suficiente. Además existen otros métodos e incluso otros MTAs (agendes de correo) como qmail o smail, que pueden realizar el mismo papel de una manera más sencilla.

Una vez realizada la configuración, supongamos que desde un usuario cualquiera enviamos un email a jose@proveedor.es, mediante el comando mail jose@proveedor.es. Dicho email se grabará en `/var/spool/mqueue` a la espera de ser enviado a nuestro ISP quien a su vez lo enviará al usuario juan. Es decir, los emails que enviemos a Internet se guardarán en el directorio del correo hasta que se realice su envio mediante sendmail en la próxima conexión a Internet, gracias al comando "sendmail -q". Los mensajes en cola de envío se encuentran en `/var/spool/mqueue` (por si queremos releerlos o borrar alguno), y podemos consultarlos mediante el comando mailq.

Sendmail es un daemon o demonio, es decir, está ejecutandose en nuestra consola continuamente tratando de enviar el correo cada X tiempo. Para evitar esto, y obligarle a que el envío se realice manualmente por nuestra parte, debemos modificar el arranque de sendmail en el fichero `/etc/rc.d/init.d/sendmail` cambiándolo para que arranque con las siguientes opciones:

```
daemon /usr/sbin/sendmail -bd -oDeliveryMode=d
```

Otro problema común es recibir una advertencia (warning) cada vez que un mensaje lleve más de X hroas sin ser enviado (4 por defecto). Para evitar eso tan sólo hay que editar /etc/sendmail.cf y comentar (colocando un símbolo # al principio de la línea, y, por tanto, deshabilitándola) la siguiente opción:

```
Timeout.queuewarn=4h
```

Otra opción a comentar (añadiendo # al principio de la línea correspondiente en sendmail.cf) es la siguiente:

```
Timeout.queuereturn=5d
```

Esta opción especifica que si el correo no ha podido ser enviado durante 5 días debe ser devuelto a sendmail. Esto puede ser util si estamos enviando los emails a un proveedor y se pierden, con lo que gracias a la devolución podrían ser reenviados, pero en nuestro caso no va a ser necesario pues los mensajes residiran en nuestro disco duro hasta el momento en que los enviemos.

En resumen: sendmail es un agente de transporte de correo (MTA) que recoge nuestros emails y los deja temporalmente como simples ficheros de texto que son en /var/spool/mqueue a la espera de realizar su distribucion a los destinatarios especificados. Esta entrega se puede hacer bien mediante el comando "sendmail -qº bien esperando a que el sistema llame a sendmail (lo llama cada X tiempo, siendo este tiempo programable) para que lo distribuya. Este tiempo, así como los restantes parámetros de configuración (como el servidor SMTP, que es el lugar a donde sendmail debe enviar los mensajes, es decir, nuestro proveedor de Internet) se pueden especificar en el fichero de configuración /etc/sendmail.cf .

Gracias a sendmail podremos enviar mensajes de corre electronico estando desconectados de Internet, mensajes que serán enviados por él cuando se realice la próxima conexión. Esto nos permitirá responder el correo electrónico desde nuestro cliente de correo, con el siguiente mecanismo: una vez configurado sendmail, en nuestro cliente de correo ponemos como SMTP la direccion IP de nuestra máquina (nuestra IP de red, nuestro nombre de máquina o bien 127.0.0.1 o localhost) de modo que cuando escribamos un mensaje en Netscape y pulsemos en Enviar, éste será enviado *a nuestra propia máquina* de tal forma que sendmail se encargará de enviarlo durante la próxima conexión a Internet cuando se ejecute mediante "sendmail -qº como daemon temporal. Esto nos permite contestar el correo electronico en nuestro cliente favorito, sin gasto de teléfono por nuestra parte, y pudiendo simplemente enviar los mensajes y dejar que sendmail los distribuya al conectar a Internet.

10.4 Recogida de correo: Fetchmail

Si con programas como sendmail es posible enviar el correo a nuestra máquina estando desconectados de Internet para su posterior distribución cuando se conecte a la Red, con Fetchmail es posible hacer el proceso inverso, es decir, recoger TODO el correo de todos los servidores POP o IMAP deseados para su posterior recogida desde nuestra máquina y lectura en modo desconectado.

Fetchmail es el daemon encargado de la recogida de correo, para posteriormente pasárselo a sendmail, que es quien lo distribuye. La forma más sencilla de configurar fetchmail es creando un fichero .fetchmailrc en el directorio personal del usuario (ya sea `/home/usuario` o incluso `/root`), de forma que sea el mismo usuario quien recoja el correo y se lo pase al usuario/usuarios que deban leerlo. Dicho fichero tiene un formato como el que sigue:

```
defaults
fetchall
flush
pass8bits
```

```
poll servidor_de_correo
proto pop3
user usuario
pass password
to usuario_local
```

(pueden haber más bloques poll/proto/user/pass/to si hay mas servidores de correo de donde deseemos recoger el mismo)

Las opciones incluidas en este fichero son las siguientes:

- defaults: establece los valores por defecto para fetchmail.
- fetchall: recoge todos los mensajes pendientes en el servidor de correo.
- flush: una vez recogidos los mensajes son borrados del servidor de correo. Para nuestras primeras pruebas es recomendable cambiar este parámetro por keep, para que los mantenga y así no los perdamos en caso de un error de configuración.
- pass8bits: Indica que el mensaje puede contener carácteres de 8 bits (acentos, eñes, etc.).
- Bloque poll/proto/user/pass: Indica las características de nuestro servidor de correo. Si disponemos de más de un servidor de correo, se deben incluir tantos bloques de este tipo (repetir la estructura para ambos) como servidores dispongamos.
- poll: Indica el servidor de correo entrante (POP o IMAP) de donde leer el correo (en mi caso se trata simplemente de anteponer pop al dominio de mi ISP, quedando pop.arrakis.es).
- proto pop3: Indica que nuestro servidor de correo entrante utiliza POP para la recogida del correo (alternativo a IMAP).
- user y pass: Indican la cuenta del correo y password del usuario. Algunas cuentas de correo típicas son como las siguientes:

  ```
  user sromero@arrakis       <- en arrakis
  user sromero               <- en CTV
  ```

  (consultar con nuestro ISP cual es el login de nuestra cuenta de correo). Si nuestro password es numérico tal vez haga falta encerrarlo entre comillas dobles .
- to usuario: Indica a qué cuenta de usuario se le ha de pasar el correo. Usuario ha de ser una cuenta de nuestro sistema Linux, tales como sromero, o root.

Un ejemplo de fichero .fetchmailrc para un usuario sromero@arrakis.es en /home/sromero sería el siguiente:

```
defaults
flush
fetchall
pass8bits
poll pop.arrakis.es
proto pop3
user sromero@arrakis
pass mipassword
to sromero
```

Además, este fichero tiene que tener unos permisos de lectura/escritura concretos, debido a que posee nuestro password de correo, cosa que se realiza con la orden chmod:

```
[sromerolocalhost sromero]#   chmod 0600 .fetchmailrc
```

La forma de recoger el correo es, estando conectado, ejecutar la orden "fetchmail".

Por último, podemos hacer que fetchmail corra en forma de demonio (el equivalente en MSDOS sería como un programa residente), y que recoja el correo sólo cada X segundos, cosa muy sencilla de realizar añadiendo la siguiente opción al fichero .fetchmailrc:

```
set daemon X

(ejemplo: set daemon 300)
```

Esto haria que fetchmail buscara correo nuevo cada 300 segundos sin necesidad de ser lanzado por nuestra parte con el comando "fetchmail".

Una vez recogidos los mensajes (ya sea en forma manual o como daemon), éstos son depositados por sendmail en `/var/spool/mail`, de manera que puedan ser accedidos por cualquier programa de correo del sistema (mailx, pine, y, como veremos ahora, Netscape). Simplemente debemos configurar nuestro cliente de correo especificando como servidor POP nuestra propia máquina (localhost, 127.0.0.1, etc). Tras haber recogido todos los mensajes con fetchmail y ya desconectados de la Red, en el cliente de correo (por ejemplo Netscape) le damos a GET NEW MESSAGES (Obtener nuevos mensajes) y aparecerán ante nosotros los emails recogidos por fetchmail dispuestos para ser respondidos totalmente offline.

10.5 Sendmail y Fetchmail: Todo Offline

Con fetchmail y sendmail, trabajar con el correo electrónico en modo desconectado (tanto leer como contestar el correo OFFLINE, con el consiguiente ahorro de dinero en tiempo de conexión) consiste simplemente en lo siguiente:

```
[root@localhost root]# pppd
[root@localhost root]# sendmail -q
[root@localhost root]# fechmail
[root@localhost root]# killall pppd
```

Con la primera línea conectamos a Internet, para posteriormente enviar el correo pendiente en la cola de mensajes (los enviados por nosotros a nuestramáquina), así como recoger el correo entrante en nuestro ISP con fetchmail (y dejarlo en nuestra máquina listo para recogerlo con Netscape, Pine o nuestro cliente de correo).

Gracias a esto en apenas unos minutos de conexión enviamos todo el correo saliente y recogemos los nuevos mensajes, dejándolos en nuestro sistema listos para su contestación con el modem desconectado. Otra opción es incluir las llamadas a las funciones de envío y recogida en el fichero /etc/ppp/ip-up, de manera que la recogida sea automática al realizarse la conexión. Además podremos utilizar procmail para distribuir el correo, de manera que todos los usuarios de una red tengan la misma dirección de email en Internet pero luego puedan enviarse mensajes a usuarios individuales con esa misma dirección, etc. Las posibilidades de actuación de Linux son infinitas en este campo, pudiendo crear listas de correo, filtros anti-spam, etc.

10.6 ¿Cómo configuro Netscape para trabajar Offline?

Si disponemos de nuestros emails en `/var/spool/mails` nada nos impide utilizar Netscape para trabajar con ellos, y así de paso trabajar en modo desconectado u offline. Para ello tan sólo hemos de ir a las opciones del programa (edit->preferences), y modificar los datos que especifican nuestro servidor de correo entrante y saliente.

Para ello vamos a la pestaña Mail & Groups, submenú Mail Server y como Outgoing mail (SMTP) server especificamos localhost (nuestro ordenador), de manera que al enviar emails con Netscape vayan a parar al directorio `/var/spool/mail`. Como Mail server type especificamos la opción "Movemail Application", y seleccionamos Built in, para que utilice la herramienta interna de Netscape para la recogida del correo, tal y como se puede ver en la figura adjunta.

Por último como root habremos de cambiar los permisos del directorio /var/spool/mail para que Netscape pueda escribir en ellos:

```
[root@localhost root]# cd /var/spool
[root@localhost spool]# chmod 01777 mail
```

(otra posibilidad es buscar la aplicacion movemail y darle permisos de root.root y atributos +s para que tome privilegios de administrador).

Una vez realizado esto, todas las recepciones y envíos se realizan sobre el directorio /var/spool/mail y /var/spool/mqueue, con lo que podremos leer y responder desde Netscape los emails recogidos con fetchmail, y cuyas respuestas serán enviadas por sendmail en la siguiente conexión a Internet. De la misma manera, podemos especificar localhost como servidor de correo entrante (pop) y saliente (smtp) en otros programas de correo para que accedan a nuestra máquina y podamos trabajar en modo desconectado, como en el programa Kmail incluido en KDE, en Pine, etc.

10.7 ¿Qué son las news o grupos de noticias?

Las news (grupos de noticias) son un interesante servicio de Internet que permite el intercambio de mensajes en un foro común sobre un determinado tema de interés para todos sus lectores. En cuanto a formato, son algo similar a los emails: simples mensajes de texto formados por la cabecera del mensaje seguida del cuerpo, y donde a su vez la cabecera se divide en diferentes campos que indican el remitente, el grupo destinatario o el tema del mensaje.

Si se busca información de un determinado tema (programación, electrónica, diseño, imágenes, etc.), sólo es necesario apuntar nuestro programa de news (por ejemplo, Collabra Discussion Groups, de Netscape, incluido con Communicator) hacia el grupo adecuado para disponer de la posibilidad de intercambio de información e intereses comunes entre un gran grupo de usuarios interesados en el mismo tema. Apuntándose, por ejemplo, al grupo es.comp.os.linux estaremos en conexión con un grupo temático dedicado exclusivamente al habla sobre este S.O. Si enviamos un mensaje al grupo, todas las personas suscritas al mismo podrán leerlo y contestarlo (algo así como enviar emails pero a un lugar donde lo pueden leer todas las personas interesadas en el tema) tanto personalmente como al grupo en general, pudiendo por tanto aprender mucho sobre el tema tratado. La variedad de temas disponibles en los diferentes grupos permiten nuestra participación activa, yendo desde la simple lectura (de la que se puede aprender mucho) hasta la colaboración con el grupo respondiendo a preguntas que ayuden a otros lectores a desarrollarse.

10.8 ¿Cómo se accede a las news?

En general, basta con instalar un cliente de news e indicarle la direccion de nuestro servidor de news (el nombre o IP de la máquina que contiene los grupos), tal como news.arrakis.es, news.ctv.es, o como quiera que se llame nuestro proveedor de servicios de Internet. También existen diferentes servidores de news gratuitos, algunos de ellos sin limitaciones y otros que sólo permiten postear mensajes de un determinado nº de líneas, o donde el nº de líneas de texto nuevo sea mayor que el texto al que se contesta, ya que al ser como un email, se puede dejar el texto anterior al contestar:

```
Ejemplo:

--------------------------------------------------
> JuanJO preguntó:
> ¿alguien sabe como hacer esto?

 Pues tienes que ir al directorio... [etc]
--------------------------------------------------
```

Sabido ya nuestro servidor de news y el puerto de acceso (general- mente el 119) la configuración del acceso simplemente consiste en ir a las opciones del cliente de news que deseemos usar e indicarle como "Servidor de News"la dirección de nuestro server, y como puerto, el apropiado para el mismo.

Una vez realicemos la configuración, el cliente de news se bajará del servidor remoto una lista de todos los grupos temáticos disponibles y de las descripciones de los contenidos, para que posteriormente podamos seleccionar aquellos de nuestro interés. Suelen haber miles o decenas de miles de grupos diferentes, ordenados por tematicas. Los grupos de news son como una jerarquía donde cada grupo se divide en subgrupos hasta llegar a aquel de la tematica deseada:

```
es.*                   -> todos los grupos de news en castellano
                          (hay miles de ellos)
es.ciencia.*           -> grupos de news sobre ciencia.
                          (hay unas decenas, de electrónica,
                           matemáticas, biología, etc)
es.ciencia.electronica -> grupo de news sobre electronica en castellano.

o bien:

es.comp.*              -> grupos relacionados con ordenadores.
es.comp.os.*           -> grupo de news dedicados a S. Operativos.
es.comp.os.linux       -> grupo de news sobre el S.O. Linux.
es.comp.os.windows     -> grupo de news sobre el "S.O." Windows
(etc...)
```

Esta lista de grupos y sus descripciones se almacena en nuestro PC para que posteriormente a su download completo (suele tardar bastante al ser muy extensa) podamos suscribirnos a aquellos grupos que más nos interesen. En el caso del autor, y como ejemplo, uso Netscape Communicator y estoy suscrito a es.ciencia.electronica, es.comp.os.linux y rec.games.programmer, los cuales recibo en diferentes carpetas para leer mensajes de cada uno de los temas por separado (cada vez que conecto a Internet recojo los nuevos mensajes de cada grupo y envio mis respuestas).

Tanto Netscape Communicator como otros lectores/clientes de noticias (Slrn, Krn, Xrn) dan acceso a la lectura y gestión de artículos de las news siempre y cuando estemos conectados a Internet (trabajando online), es decir, hay que estar conectado, recoger los nuevos mensajes, y leerlos y contestarlos estando conectados lo cual conlleva un gasto (innecesario como veremos) de teléfono a la hora de leer, responder y descartar los mensajes que nos interesen.

Existen algunos programas que permiten hacer algo más ahorrativo: al entrar en el grupo de news que nos interesa, podremos leer las cabeceras de los mensajes disponibles (como por ejemplo en el programa

HY-News), marcar los que nos interesen y posteriormente bajarlos para leerlos una vez desconectados de la línea, si bien el marcado de cabeceras requiere un tiempo de lectura y elección que puede llegar a bastantes minutos si estamos suscritos a diferentes grupos, aunque siempre es mejor esto que responder todos los artículos online.

En Linux disponemos de la posibilidad (de nuevo formando parte gratuita del Sistema Operativo y no como un extra a añadir al mismo) de instalar clientes de news (y servidores) que nos permitirán trabajar en modo desconectado de la misma manera que se puede hacer con el correo (en aquel caso mediante fetchmail y sendmail). Estos programas funcionan bajandose TODOS los nuevos articulos/mensajes de news (a gran velocidad al ser conexion directa al servidor) para luego en nuestro cliente de correo especificar como servidor de news nuestro propio PC (localhost) de modo que podremos leer y contestar las news totalmente offline.

En este aspecto podemos elegir principalmente entre leafnode e inn+suck. Éstos 2 ultimos paquetes (inn y suck) son muy potentes y mucho más avanzados para nuestros propósitos iniciales, de modo que si nuestro uso de las news es sencillo (no queremos montar nuestro propio servidor sino simplemente obtener un medio para leer las news offline) nos quedaremos con leafnode mucho más sencillo de instalar y configurar.

10.9 Leafnode como servidor de NNTP

Como puede leerse en la página del manual (man fetch) leafnode es un paquete de trabajo con news diseñado para ordenadores pequeños con pocos usuarios, de manera que no ocupa mucho espacio y permite la gestión de muchos grupos. El diseño de leafnode está pensado para que sea capaz de autoreparar errores cuando ocurran, y que no necesite mantenimiento manual. Es decir: nos encontramos ante un programa que requiere poco espacio y mantenimiento, pero que en cambio nos proporciona total acceso a la gestión de grupos de noticias, gracias a leafnode (un servidor de NNTP), fetch, el programa que se encarga de recoger las news de Internet y dejarlas en nuestro disco duro para su posterior lectura por parte del programa que deseemos, y texpire, que se encarga de eliminar los mensajes viejos para recuperar disco duro y deshacernos de los artículos no deseados según el tiempo de expiración que hayamos configurado.

Leafnode es un programa incluido en la mayoría de las distribuciones Linux (sobre todo en las de RedHat), y si no disponemos de él puede encontrarse en ftp.redhat.com en el directorio pub/manhattan/contrib/i386 (versión 1.5) o en la sección de viejos RPMs (la versión 1.4). Algunas (caso de algunas distribuciones Manhattan) versiones tienen un fallo con Netscape, pero al autor le consta que a partir de la versión 1.9 este fallo fue corregido completamente (NOTA: El autor usa leafnode 1.4 ya que le sigue funcionando perfectamente y no ve ninguna necesidad de actualizar a la version 1.9 :). La instalación en formato rpm es mediante el comando rpm -i, y para paquetes .deb y tar.gzs también se debe seguir el método habitual de instalación ya conocido por los que usan distribuciones Debian o Slackware, además de poder convertir el paquete de rpm a cualquier otro formato mediante el script alien, incluido en muchas distribuciones y que podemos encontrar en el home de Debian.

10.10 Configuracion de Leafnode

Una vez instalado leafnode se debe proceder a configurarlo. Como root entramos en el directorio /usr/lib/leafnode y editamos el fichero config, donde leeremos algo similar a lo siguiente (fichero de configuración de la versión 1.4):

```
# This is the NNTP server leafnode fetches its news from.
# You need read and post access to it.
server = news.arrakis.es

# Unread discussion threads will be deleted after this many days if
# you don't define special expire times. Mandatory.
```

```
expire = 5

# Non-standard expire times (no regex possible)
#groupexpire comp.linux.misc = 5   # group too big to hold articles 20 day
#groupexpire any.newsgroup = 100  # very interesting, hold articles longe

# Never fetch more than this many articles from one group in one run.
maxfetch = 2000

# Specifies an upper limit on how many groups an article may be posted to.
maxcrosspost = 0
```

El significado de las diferentes opciones es muy descriptivo:

La opción server indica el servidor de donde leafnode ha de leer las noticias. En este caso se tiene especificado como servidor news.arrakis.es. En principio esta línea no contendrá ningún valor si es la primera vez que lo configuramos, así que será necesario incluir aqui el nombre de nuestro servidor de noticias habitual (news.ctv.es, news.arrakis.es, etc.).

Con expire=5 le indicamos a leafnode que si un tema (thread o hebra) de artículos no ha sido tratado durante 5 días o más, debe borrarlo (lo haremos con texpire). Es decir, si un hipotético tema llamado «Pregunta simple», y sus respuestas «Re: Pregunta simple» llevan 5 días sin ser tratados, todos estos mensajes son eliminados del disco duro para ahorrar espacio (debido a que es considerado como tema antiguo). El número de días puede ser modificado a nuestro gusto mediante esta opción.

Las 2 opciones groupexpire comentadas permiten especificar excepciones a diferentes grupos (por ejemplo, si queremos que en un determinado grupo se guarden los artículos más días) con respecto al tiempo considerado para eliminar artículos. En este sentido, se utiliza para hacer una excepción para un grupo concreto sobre el valor de expire generalizado para todos los grupos (por ejemplo, si un grupo tiene un flujo muy grande de mensajes puede no interesarnos que caduquen a los 5 dias sino a los 3).

El parámetro maxfetch especifica el máximo número de artículos a recoger cada vez de cada grupo. La primera vez que recojamos los mensajes de un grupo nos recogerá este número de posts, así que es recomendable ponerlo a 500 inicialmente y despues de la primera recogida de grupos dejarlo en el valor deseado.

La opción maxcrosspost indica el número máximo de grupos a los que un artículo puede ir dirigido (para evitar el crossposting o repeticion del mismo mensaje en diferentes grupos), o cero para no indicar límite de grupos.

Una vez configurado el programa conectamos a Internet y ejecutamos el programa de recogida de news (fetch) mediante la orden:

```
[root@localhost root]#   fetch -v
```

Mediante esta orden, y al ser la primera vez que llamamos al programa, fetch se dedicará a bajarse de nuestro servidor de news todos los nombres y descripciones de los grupos de noticias existentes para que posteriormente podamos suscribirnos a cualquiera de ellos.

Si queremos seguir el proceso y saber que hace en todo momento leafnode, podemos ejecutar el siguiente comando en otra consola virtual o xterm:

```
tail -f /var/log/messages
```

Veremos algo similar a:

```
Jan 2 11:07:00 localhost fetch: Registered group arrakis.binarios
Jan 2 11:07:00 localhost fetch: Registered group arrakis.consultas
Jan 2 11:07:00 localhost fetch: Registered group arrakis.general
Jan 2 11:07:00 localhost fetch: Registered group arrakis.noticias
Jan 2 11:07:00 localhost fetch: Registered group a.bsu.programming
Jan 2 11:07:00 localhost fetch: Registered group a.bsu.religion
(y un largo etc).
```

El parámetro -v (verbose) le indica a leafnode que muestre mensajes en pantalla indicando qué hace en cada momento. Hay diferentes niveles de explicación (-v, -vv, ...-vvvvv) que nos permitirán saber qué hace el programa y observar la progresión del mismo. Tras un tiempo (dependiendo de la velocidad de nuestro modem), el programa terminará de recoger los nombres de los newsgroups (proceso largo, debido a la gran cantidad de ellos, así que tendremos que tener paciencia esta primera vez de ejecución), y habremos de resetear el inetd para que se active nuestro servidor de news (leafnode):

```
[root@localhost root]#   /etc/rc.d/init.d/inet  restart
```

Por último (aunque suele ser realizado automáticamente al instalar el paquete), debemos asegurarnos de que tenemos abierto el puerto de nntp (de nuestro servidor de noticias) en el archivo /etc/inetd.conf, simplemente comprobando si existe (y creando en caso negativo) una línea similar a la siguiente:

```
nntp  stream  tcp  nowait  news  /usr/sbin/tcpd  /usr/sbin/leafnode
```

Tras esto podemos desconectar de Internet y prepararnos para elegir los grupos a los que deseamos suscribirnos, como veremos a continuación.

Nota: Es posible que si instalamos leafnode desde un fichero tar.gz, el path destino de instalacion sea /usr/local/sbin/leafnode, de modo que la linea anterior debe cambiar para ajustarse al path donde leafnode este disponible.

```
nntp  stream  tcp  nowait  news  /usr/sbin/tcpd  /usr/local/sbin/leafnode
```

Las ultimas versiones llevan un mayor número de opciones las cuales podemos (y deberíamos) consultar en la página man de fetch.

10.11 Suscripcion a los grupos de noticias deseados

Una vez tenemos la lista de grupos y sus descripciones (Active List) en nuestro disco duro, llega el momento de suscribirse a los grupos que deseamos recoger. Para ello nos vamos a /var/spool/news/interesting.groups y creamos allí ficheros con los nombres de los grupos a los que nos queremos suscribir. Dicho de una manera sencilla, si nos queremos suscribir a es.comp.os.linux y es.ciencia.electrónica, por ejemplo, dentro de /var/spool/news/interesting.groups creamos dichos ficheros sin contenido alguno:

```
[root@localhost root]#   touch  es.comp.os.linux
[root@localhost root]#   touch  es.ciencia.electronica
```

El comando touch sirve para actualizar la fecha de un fichero, y si no existe se encarga de crearlo (vacío, de cero bytes). El directorio interesting.groups es para leafnode una base de datos de los grupos que debe bajarse, estando especificados en forma de ficheros. Después de los 2 touch, el contenido del directorio es el siguiente:

```
[root@localhost interesting.groups]# ls -l
total 0
-rw-r--r--   1 news     news     es.ciencia.electronica
-rw-r--r--   1 news     news     es.comp.os.linux
```

Mediante esto, cada vez que recojamos los artículos llamando a fetch (como ahora veremos), leafnode se bajará los artículos nuevos de estos 2 grupos. Por supuesto, es posible suscribirse a tantos grupos como sea necesario, creando los ficheros con touch (o creandolos con cualquier editor de texto y grabándolos vacíos).

Tras esto ya estamos suscritos a diferentes grupos en el leafnode, lo cual significa que leafnode se bajará los mensajes de dichos grupos. El siguienet paso sería ir al cliente de news y poner como Groups Server (servidor de news, de nntp o de grupos) el nombre o IP de nuestra propia máquina, como 127.0.0.1 o localhost. Tras esto veremos aparecer en el cliente de news (por ejemplo en Netscape Communicator) la lista de grupos que hay en nuestro sistema (la que se bajó leafnode) y nos podremos suscribir a los grupos que deseamos leer en nuestro netscape (Join Discussion Groups).

Es decir, primero con Leafnode nos suscribimos a los grupos que deseamos recibir en el sistema, y luego cada usuario, en su cliente de news, se suscribe, de dichos grupos que se reciben, a los que desea leer en su cliente.

10.12 Recogida de news

Una vez configurado todo el sistema de leafnode, veamos el proceso que se seguiría cada vez que se deseen recoger los artículos de las news cada día (o cada vez que se use Internet). Para ello, conectamos a Internet (por ejemplo, ejecutamos pppd, enviamos nuestro correo con sendmail, recogemos los mensajes nuevos con fetchmail, etc.) y ejecutamos el siguiente comando:

```
[root@localhost root]#   fetch -v
```

En ese momento, leafnode enviará los artículos que hayamos respondido (si es la primera vez, no habremos escrito nada todavía) y después comenzará a bajarse todos los artículos nuevos de los grupos a los que nos hayamos suscrito en `/var/spool/news/interesting.groups`. Nótese que leafnode la primera vez se bajará unos 2000 mensajes de cada grupo (depende de la opción maxfetch especificada en el fichero de configuración, por lo que recomendamos que inicialmente se deje en 500 para colocarla en 2000 tras la primera recogida), pero a partir de ese momento en la siguiente conexión sólo se bajará los artículos nuevos.

Nótese también que leafnode se baja todo el grupo (todos los mensajes, no siendo necesario marcar cabeceras), con lo que no es necesario pasar tiempo marcando los artículos deseados, sino que posteriormente dispondremos de todo el grupo para su lectura en offline. Además leafnode es muy rápido bajando los artículos, pues lo hace a altas velocidades y permite también ser añadido al script ip-up para su recogida automática tras la conexión.

Una vez recogidos los artículos estos son dejados en `/var/spool/news` (en nuestro disco duro) para que puedan ser accedidos por cualquier cliente de news, como Krn, Xrn, Netscape, etc. Por otra parte, una vez por semana leafnode realiza una actualización de la lista de grupos y descripciones (no recogida completa sino actualización) aunque esto se puede evitar si así lo deseamos, como veremos más adelante.

Otra de las ventajas de que la recogida se haga por medio de un comando de consola es que podemos colocar en un script de shell (como ya dijimos, algo similar a un BATCH de MSDOS, pero mucho más potente), que nos haga la recogida automática de correo y news para nuestro ordenador. Para ello creamos un fichero "recoge"con el siguiente contenido:

```
# Script para recogida de correo
echo Enviando correo pendiente...
sendmail -q
```

```
echo Recogiendo correo entrante...
fetchmail
echo Posteando y recogiendo news...
fetch -v
```

Tras crear este fichero, le damos permisos de ejecución (chmod +x recoge), y ya puede ser utilizado en alguna consola virtual o xterm para recoger el correo+news automáticamente (enviar y recoger correo y news), mientras navegamos o trabajamos con el resto del sistema.

10.13 Configuracion del cliente de news para trabajar con leafnode

La configuración del cliente que utilicemos es muy sencilla, y consiste en cambiar el servidor de news que tuvieramos especificado en las opciones del programa (por ejemplo, news.arrakis.es) por nuestra propia máquina (localhost), con lo que cuando sea pulsado el botón de "Coger nuevos mensajes", por ejemplo, se estará accediendo a nuestra máquina (concretamente a leafnode) para leer dichos nuevos mensajes de `/var/spool/news`, es decir, los artículos recogido por fetch.

En Netscape Communicator esto se especifica en el menú Edit, opción Preferences, pestaña Mail & Groups, subopción Groups Server, donde especificaremos 127.0.0.1 (o localhost) como Groups Server y nuestro home (`/home/sromero` en mi caso) en la opción Discussion Groups (news) directory. Tras eso nos suscribiremos a los grupos deseados (cada usuario puede elegir cualquier grupo de los especificados en interesting.groups) mediante la opción Join Discussion Groups del menú File. El proceso es similar para otros clientes como Krn, Slrn (con un sencillo fichero de configuración) o incluso el editor de texto Emacs (que puede utilizarse también para gestionar las news).

Una vez realizado esto, al pulsar sobre "Get new messages"estaremos accediendo a nuestro disco duro para leer los ultimos artículos recogidos por leafnode, y al enviar un mensaje en realidad estará siendo grabado como un fichero de texto en el directorio /var/spool/news/out.going, donde se almacenan todos los mensajes enviados offline hasta que en la próxima conexión fetch los envíe a nuestro servidor de news (y por tanto, sean posteados en Internet).

10.14 Eliminacion de noticias antiguas (Texpire)

Cada cierto tiempo debemos limpiar nuestro disco duro para eliminar mensajes antiguos. Esto no significa que debamos entrar en el directorio news para comprobar qué mensajes están atrasados y eliminarlos, porque como ya se ha comentado, leafnode es un programa diseñado para automatizar todo este tipo de procesos. Para ello sólo tendremos que ejecutar el programa texpire (incluido con leafnode), que se encarga de recorrer todos los grupos de que dispongamos en nuestro disco duro y eliminar aquellos threads que sobrepasen sin actualizar el número de días indicado en el fichero de configuración de leafnode (en nuestro ejemplo indicamos 10 días). Al ejecutar texpire la salida suele ser algo parecido a lo siguiente (la aparición de varios grupos a los que no estamos suscritos es debido a que hay mensajes con crossposting, dirigidos a varios grupos):

```
[root@localhost root]# texpire
es.ciencia.electronica: 10 articles deleted, 182 kept
es.ciencia.misc: 21 articles deleted, 25 kept
es.comp.hackers: 1 articles deleted, 10 kept
es.comp.lenguajes.java: 10 articles deleted, 1 kept
es.comp.os.linux: 100 articles deleted, 690 kept
total: 142 articles deleted, 1012 kept
```

Realizando esto regularmente mantendremos nuestro disco duro sólo con los últimos mensajes deseados en lugar de ir llenándolo con la acumulación de mensajes antiguos. La cantidad de días a los que hayamos configurado texpire variará el nivel de limpieza de artículos viejos. Texpire es automáticamente ejecutado por "cron"(la parte de Linux que se encarga de ejecutar procesos a determinadas horas), pero para que cron tenga utilidad es necesario tener conectado el ordenador todo el dia (cron hace las cosas a unas determinadas horas), cosa que viene muy bien en las estaciones de trabajo pero que a nosotros no obliga a usar el ªnacron"(no necesita tener el ordenador permanentemente conectado) o a hacerlo manualmente (la forma más sencilla, simplemente llamando a texpire cada 15 días, ejecutándolo en la línea de comandos del sistema).

Normalmente a texpire le costará cerca de un mes de uso empezar a borrar artículos, de modo que aunque inicialmente veamos que al llamarlo no borra los artículos antiguos, debemos dejarlo porque eso quiere decir que ninguno de los artículos ha expirado aún (o que hemos puesto un tiempo de expiracion muy grande. Un tiempo razonable es 5 dias, pero eso depende fundamentalmente del tráfico del grupo en cuestión). Además texpire está basado en detectar TEMAS caducados (no artículos). Esto implica que un TEMA caduca cuando no se postee ningún mensaje sobre él (o respuesta) en el tiempo especificado, tras el cual será borrado. Estar basado en temas hace que se tarde más (inicialmente) en comenzar a borrar artículos viejos, pero que a partir de la primera expiración de artículos el proceso sea contínuo y eficiente.

Por Internet es posible encontrar también un programa en PERL llamado texpire.pl el cual hace un borrado artículo por artículo (no basado en cuando un TEMA caduca sino en cuando un ARTICULO caduca) que también puede utilizarse para este fin. No obstante el uso de texpire es más adecuado ya que viene integrado con leafnode y su funcionamiento es excelente.

10.15 ¿Cómo hacer que leafnode no actualice la lista de grupos?

Por otra parte, si no queremos que leafnode trate de actualizar los grupos de news cada semana es posible modificar el código fuente del programa, aunque lo más sencillo es actualizar la fecha del fichero de grupos para que leafnode lo trate como si se acabara de actualizar. Esto se hace mediante la siguiente orden, que podemos poner en cualquier fichero del sistema (por ejemplo en /etc/rc.d/rc.local):

```
touch -m /var/spool/news/active.read
```

Este fichero (active.read) lo usa leafnode para saber cuándo fue la última vez que buscó nuevos grupos de news en el servidor. Cuando pasa una semana (fetch mirará para ello la fecha del fichero), leafnode tratará de bajarse de nuevo la lista de grupos para ver si hay grupos nuevos (proceso que suele tardar bastante). Para evitar esto, touch actualiza la fecha del archivo y fetch nunca detecta que haya pasado una semana, no actualizando la lista de grupos y descripciones.

En las últimas versiones, posiblemente podamos forzar a leafnode a no leer la lista de grupos o a releerla de nuevo usando algún switch de línea de comandos (al estilo fetch -n), para lo cual siempre es aconsejable la lectura de la página man correspondiente (man fetch).

10.16 Problemas de acceso con Leafnode

Si al recoger las news con Netscape o cualquier otro cliente de news obtenemos el siguiente mensaje de error:

```
"An error occurred with the News server.
If you are unable to connect again, contact the
administrator for this server."
```

Lo primero que deberemos hacer será comprobar si leafnode está correctamente funcionando en el puerto 119 de nuestra máquina, haciendo un telnet localhost 119:

```
[root@compiler]  telnet localhost 119
```

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Connection closed by foreign host.
```

En ese caso, debemos modificar los ficheros `/etc/hosts.allow` y `/etc/hosts.deny` para permitir a nuestros usuarios que tengan acceso a leafnode. Una manera es dejar vacíos ambos ficheros (así es como lo tengo yo) permitiendo a cualquier el acceso a leafnode, y otra es modificar el hosts.deny (nunca he necesitado realizarlo así pero es otra manera de solucionarlo) es dejar el siguiente contenido en el fichero `/etc/hosts.deny`:

```
# /etc/hosts.deny
# The PARANOID wildcard matches any host whose name does not
# match its address.
ALL: PARANOID
#-- leafnode begin
leafnode: ALL
#-- leafnode end
```

Una salida de "telnet localhost 119"correcta debería ser:

```
[root@compiler]  telnet localhost 119
Trying 127.0.0.1...
Connected to localhost
Escape character is '^]'.
200 Leafnode NNTP Daemon, version 1.4 running at localhost
```

11 IRC

11.1 ¿Qué es el IRC?

El IRC, acrónimo de Internet Relay Chat, es un sistema de conversación en línea para usuarios de Internet Si disponemos de una conexión a Internet que ofrezca este servicio (cosa muy común en todos los distribuidores de Internet), será posible disfrutar de charlas en tiempo real sobre cualquier tema (de entre los cientos e incluso miles de canales que tendremos disponibles) con una gran cantidad de usuarios.

Para ello tan sólo necesitaremos un cliente de IRC, como ircII (incluido con Linux en casi todas las distribuciones) o, como veremos, con variantes más visuales como BitchX, Zircon, cIRCus o Kirc. Mediante la simple ejecución de nuestro cliente IRC entraremos en nuestro servidor de IRC (previa configuración del programa cliente para que busque en la dirección correcta) y se nos dará la posibilidad de entrar en cualquiera de los canales (channels, agrupaciones temáticas), descritos por sus títulos (topics o títulos «del dia») donde encontraremos otros usuarios y algunos operadores (ops). También existe la posibilidad de hablar en privado con otro usuario, de enviar y recibir ficheros (vía DCC) así como unos usuarios especiales (que llevan una arroba (@) delante de su nombre) llamados operadores del canal (que han sido nombrados operadores por estar registrados en el canal o por otros operadores), que suelen estar ahi para controlar que la conversación gire en torno al tema establecido en el canal, aunque por desgracia en algunas ocasiones (en determinados canales) se dediquen a tirar gente del canal (acciones conocidas como ban o kick).

Los programas clientes de IRC suelen constar de una celdilla de texto en la parte inferior de la pantalla donde el usuario (nosotros) introduce el texto que tras pulsar ENTER es enviado al canal, apareciendo en la ventana de la conversación (prácticamente el resto de la pantalla), donde se pueden leer las frases escritas por todos los contertulios del canal en el orden en que éstas llegaron al servidor. Esto permite tener conversaciones, disputas y cooperaciones entre los diferentes miembros del canal (a veces apareciendo en una ventana en la parte derecha de la pantalla).

Mediante lo que el usuario escribe (que leen los demás) y mediante lo escrito por los demas (leído por nosotros), se forma una discusión que va desde las conversaciones sobre linux en el canal #linux hasta las discusiones sobre DirectX u OpenGL en el canal #programacion.

El aspecto de una conversacion en el IRC es similar al siguiente:

```
-----------------------------------------------------------
TALKING on #programacion:
<Gans> y yo, y yo
<blad> ese log es oro
<Gans> yo quiero que volverá para despedirse
<iXnay_> perar que lo pase a tos los que quieran XDD
<Surf_> yo mismo ixnay
<iCorreca_> bueno, me voy pa #linux :-)
<iCorreca_> alli nos vemos... :-)
<iXnay_> esperar un poco

>>> _
-----------------------------------------------------------
```

El IRC constituye pues un medio más rápido de comunicación (y por tanto muy útil) que las news o el email, donde necesitaremos algunas horas (tal vez 1 ó 2 días) para recibir respuesta a nuestra pregunta Lo primero que debemos averiguar es el nombre de nuestro servidor de IRC, tal como lo es irc.arrakis.es, el servidor irc de arrakis (consultar con nuestro proveedor de Internet). La mayoría de IRCs de España están interconectados dando lugar a lo que se conoce como el IRC Hispano, que será nuestro punto de encuentro con otros usuarios del IRC Español

Si entre nuestras necesidades de utilización de Internet está la de mantener conversaciones (escritas) con varios interlocutores organizándolas por canales aludiendo a diferentes temas, entonces el Internet Relay Chat (IRC) es nuestro servicio.

11.2 El cliente de IRC IRCII (o EPIC)

IrcII es uno de esos programas pequeños y sencillos que proporcionan gran efectividad para su tarea, tal y como lo es ftp para la transferencia de ficheros. IrcII es un cliente de IRC que de nuevo es casi parte del S.O. Linux, ya que lo encontraremos en cualquier distribución de GNU/Linux existente. Muchos de los que están leyendo esto probablemente habrán utilizado ya el IRC mediante el programa MIRC (o PIRCH) de Windows95... pues hay que decir ante todo que IRCii es la antítesis de MIRC; es decir: ircII es en modo texto, en principio no soporta colores, y todo funciona a base de comandos (aunque con la posibilidad de crear menúes y alias), pero es uno de esos programas que resultan esenciales, sobre todo en sistemas pequeños (como la gente que se lleva un diskette de arranque de Linux con soporte para redes y con el ircII para disfrutar del IRC en las Universidades, donde no todo el mundo dispone de una cuenta Unix, o donde no se suele permitir la instalación de clientes de IRC ni juegos de rol). Lo más probable es que ya se disponga de él en el sistema, aunque siempre puede buscarse en el ftp de programas Linux habitual

La instalación en formato rpm o deb no necesita comentarios, mientras que para los ficheros .tar.gz suele haber (una vez descomprimidos con tar xvzf nombre_fichero) un script de instalación (easyinst) que realizará el proceso de copia de los ficheros a sus directorios correspondientes. Una vez instalado, y si estamos conectados a Internet, basta con llamar al cliente de IRC ircII para tratar de conectar con un servidor de IRC, mediante la invocación del ejecutable irc:

```
[root@localhost sromero]# irc
```

O mejor, especificando el nombre que tendremos dentro del IRC (nickname) y el servidor a utilizar:

```
[root@localhost sromero]# irc SuperJuan irc.encomix.es
```

Al ejecutarlo tratará de conectarnos (si no le hemos especificado servidor en la línea de comandos) a cualquiera de los servidores de IRC que tiene establecidos por defecto (y que podremos cambiar editando el fichero `/usr/lib/irc/ircII.servers`), aunque también es posible especificarlo manualmente (tras ejecutar irc) mediante la siguiente orden:

```
/server irc.ctv.es
```

Server es uno de los múltiples comandos de ircII (también disponibles en el resto de clientes de IRC, por supuesto), mediante el cual le indicamos al programa que conecte con un servidor determinado. El hecho de llevar una barra (/) antes del comando le indica al IRC que estamos tratando de ejecutar el comando server en vez de tratar de escribir la cadena "server irc.arrakis.es"en pantalla).Esto prentende ser una sencilla introducción a los distintos comandos del IRC, de manera que se puedan utilizar estos en cualquiera de los clientes de IRC visuales (con ventanitas y menúes) que podamos tener instalados en el sistema.

11.3 Como entrar en canales del IRC

Una vez conectados al IRC ya es posible entrar en un canal de conversación. Es posible listar los canales del IRC con el comando /list, aunque no es recomendable debido a la gran cantidad de ellos. Por contra es mejor conocer los canales a los que queremos entrar (como #linux, #programacion, #programacion_d_juegos, #hacking, #emuladores, o, si tenemos problemas, #ayuda_irc). Para entrar en un canal se utiliza el comando /join seguido del canal al que se desea entrar:

```
/join  #linux
```

Tras ejecutar esto, nos encontraremos dentro de un canal muy poblado hablando sobre el mejor Sistema Operativo existente hasta el momento. Cualquier frase que tecleemos en este momento (es decir, cualquier cosa que no sea un comando, precedido por '/'), será enviado al canal y leido por el resto de usuarios del canal. Si es nuestra primera sesión de IRC, mejor algo de nettiquete y una sencilla presentación y saludo al canal, seguido de cualquier duda o ayuda que se quiera dar al resto de usuarios. Para saber quien está conectado en un determinado canal, entramos en él y ejecutamos el comando /who * (aunque también puede hacerse mediante el comando /names #canal, en este caso, /names #linux) Si en cualquier momento deseamos salir del canal, basta con ejecutar el comando /part (o /leave) para abandonar el canal actual, (o part #canal) ya que, como veremos a continuación, no se está limitado a hablar en un sólo canal temático

11.4 Multiventana con IRCII

El IRC no está diseñado para hablar en un único canal. Mediante /join podemos entrar en más de un canal, aunque en principio los textos provenientes de uno u otro se mezclarán en la misma pantalla (con una indicación acerca de cuál es el canal de procedencia), además de que para hablar en uno u otro canal se necesitará utilizar el comando /msg (con lo cual dirigiremos los mensajes a un canal u otro):

```
/join #linux
/join #emuladores
/msg #linux Hola a los usuarios de Linux!
/msg #emuladores Hola! Ha salido algun emulador hoy?
/msg pepe Hola Juan!
```

Con IRCii no estamos obligados a leer todo el texto en la misma ventana, sino que es posible crear diferentes ventanas virtuales (escondidas, de manera que sólo vemos una en pantalla, o partidas en la misma pantalla), mediante el comando /window new, y cerrar la ventana actual mediante /window kill. Podemos cambiar entre ventanas mediante las teclas `Ctrl+x p` (ventana anterior o previous, mediante `Ctrl+x` simultáneamente, soltar, y luego pulsar 'p') y `Ctrl+x n` (siguiente o next). Veamos como cambia la cosa:

```
/join #linux
/window new
/join #emuladores
(ctrl+x p -> estamos en #linux)
Hola a los usuarios de Linux!
(ctrl+x n -> estamos en #emuladores)
Hola! Ha salido algun emulador hoy?
/window new
/query Juan
Hola Juan!
```

Tras esto dispondremos de 3 ventanas entre las que podremos cambiar en cualquier momento, una en la que se habla en #linux, la de #emuladores, y una tercera donde se mantiene una conversación privada con Juan (mediante el comando /query). Mediante /window hide es posible hacer una ventana oculta (en vez de partir la ventana actual). Es muy recomendable tener instalado el paquete screen para el soporte multiventana, o trabajar en xterms, donde este soporte también se da. De otro modo, el soporte multiventana se dará en pantalla partida (tantas secciones como ventanas dispongamos).

11.5 Resumen de comandos del IRC

Entre los comandos más habituales del IRC están aquellos que podemos ver en el cuadro siguiente (donde lo denotado entre corchetes [y] especifica que dicho parámetro es opcional).

```
/help [comando]            Proporciona ayuda en línea sobre el
                           comando especificado.

/join #canal               Entra en el canal especificado
                           Ej: /join #linux).

/list [#cadena]            Lista los canales disponibles en el
                           servidor, con la posibilidad de especificar
                           una cadena de búsqueda. (#cadena).

/whois [usuario]           Proporciona datos acerca del usuario que
                           se le especifique, como su nombre (si configuró
                           correctamente su cliente de IRC), dirección
                           email o dirección IP.
                           Ej: /whois NoPito

/me [frase]                Indica al resto del canal lo que piensas,
                           resaltándolo. Es el responsable de leer frases como:
                           «_Juan_ piensa que esta conversación es muy tonta.».
                           (Ejemplo: /me esta aburrido!).
```

```
/part [#canal]              Abandona un canal (o el canal actual si no se
                            especifica otro).

/msg [usuario] [mensaje]    Envía el mensaje especificado al usuario que se le
                            indique (ej: /msg juan Hasta luego!).

/dmsg [usuario] [mensaje]   Similar a msg, pero para un chat dcc.

/query [usuario] [mensaje]  Permite iniciar una conversación privada
                            (un query) con otro usuario.
                            Ej: /query NoPete

/query                      Cierra la conversación privada actual.

/nick [nombre]              Permite cambiar nuestro nickname al especificado.

/dcc send [usuario] [fichero]
                            Permite el envío de un fichero al
                            usuario especificado
                            Ej: /dcc send Juan /etc/dosemu.conf

/dcc get [usuario] [fichero]
                            Permite recibir un fichero de un usuario.

/dcc chat [usuario]         Permite iniciar un chat (conversación directa)
                            con un usuario.

/dcc close chat [usuario]   Cerrar el chat previamente abierto.

/quit [razon]               Sale del programa y, opcionalmente, muestra un mensaje
                            con la razón del abandono
                            Ej: /quit Me tengo que ir!

/away [razon]               Avisa al resto de usuarios de que nos vamos a ausentar
                            momentáneamente del IRC.
                            Ej: /away El telefono!

/invite [usuario] [#canal]  Invita al usuario especificado al canal que se le indica.

/who *                      Muestra los nombres de los usuarios del canal actual.

/clear                      Limpia la ventana de texto actual.

/date o /time               Muestran la fecha u hora actual.

/exec -out [comando]        Ejecuta el comando especificado (por ejemplo, ls),
                            y saca los resultados por la pantalla del canal.

/flush                      Hace parar la salida obtenida desde el servidor.
                            Muy util para teclearlo, por ejemplo, ante un /list.
```

```
/ignore [usuario] [tipo_de_mensaje]:
                             Ignora los mensajes enviados por determinados
                             usuarios (es decir, no aparecen sus frases en
                             pantalla). En él usuario pueden especificarse
                             comodines (*), mientras que el tipo de mensaje
                             puede ser ALL (todos), NOTICES (notificaciones),
                             PUBLIC (conversación normal del canal),
                             INVITES (invitaciones) o NONE (ninguno).
                             Ejemplo: /ignore Pepe ALL -PUBLIC
                             (ignorar todos los mensajes de Pepe excepto
                             (-) los que vayan al canal (publicos)).

/leave [#canal]              Deja el canal especificado o el actual si no
                             se especifica ninguno.

/mode                        Permite a los operadores cambiar modos (ver /help mode).

/names [#canal]              Muestra los nombres de los usuarios del canal
                             especificado (o de todos si no se especifica).

/notice [#canal] [mensaje]   Envía un mensaje privado a todos los usuarios
                             de un canal.

/notify [usuario]            Permite especificar una lista de usuarios para que al
                             entrar al IRC se nos avise de si estos están presentes,
                             o de si entran o salen del IRC (para encontrar siempre al
                             amiguete de turno). Ej: /notify pepito.

/ping [usuario]              Envía un ping (una señal con la hora actual) a un
                             usuario para que este la devuelta y podamos saber (por el
                             tiempo transcurrido) la velocidad de respuesta de la red
                             (si está lenta, rápida...).

/save [opciones] [fichero]   Salva las opciones actuales del IRC en un fichero
                             (ver /help save).

/say y /send +texto          Imprime el texto especificado en el canal, permitiendo
                             alias, binds, etc.

/set log on                  Permite guardar en un fichero de texto lo hablado en
                             las sesiones.

/set logfile [fichero]       Permite especificar el fichero a usar
                             (ej: /set logfile /home/somero/milog.txt).

/window log on               Permite guardar en un fichero de texto lo hablado
                             en la ventana actual.

/window logfile [fichero]    Permite especificar el fichero a usar
```

```
                              (ej: /set logfile /home/somero/milog.txt).

/alias [comandos]             Crea un alias para [comandos].
                              También permite parámetros.
```

Aparte de los comandos especificados, otro comando muy útil es /alias, que nos permitirá crear atajos a órdenes habituales:

```
/ALIAS  Hola  MSG #linux Hola a todos los usuarios del canal!
/ALIAS  Hola  MSG  $0  Hola, $0! $1
```

Para este último alias, tras ejecutarlo, si tecleamos «/Hola Juan Que tal?», será expandido a «/msg Juan Hola, Juan! Que tal?», con lo que dicho saludo aparecerá en la pantalla de Juan como mensaje privado dirigido a él. Otros comandos interesantes (ver lista) son los comandos de DCC, que nos permitirán enviar y recibir ficheros a/de otros usuarios del IRC El resto de comandos del IRC (y una ayuda y descripción de todos ellos), puede encontrarse en el propio programa (/help comando), o en el subdirectorio `/usr/lib/irc/help`, de donde cuelga toda una estructura de directorios con ficheros de ayuda en texto puro (editables con cualquier procesador de textos o listables con cat o less) con el nombre de los distintos comandos. Todas estas opciones se suelen incluir en scripts (similares a ficheros de comandos) que son cargados durante el arranque y añaden nuevas opciones y funcionalidades al programa. Otra referencia es `/usr/doc/irc*`, donde hay ejemplos de menúes, la página man de ircII y los cambios que han ido habiendo en el programa desde su versión inicial

11.6 Resumen de teclas de IRCII

Las teclas más utilizadas en ircII son las siguientes:

```
Ctrl+x p      :   ir a la ventana virtual anterior
                  (o comando /window previous).
Ctrl+x s      :   ir a la ventana virtual siguiente
                  (o comando /window next).
Ctrl+x [nº]   :   abrir N ventanas
                  (o comando /window add N>.
Ctrl+x c      :   crea una nueva ventana
                  (o comando /window add).
Ctrl+x a      :   va a la última ventana.
Ctrl+x k      :   cierra la ventana actual.
Alt+p         :   subir por la pantalla actual.
Alt+n         :   bajar por la pantalla actual.
Ctrl+b        :   escribir en negrita.
Ctrl+_        :   escribir en subrayado.
```

Además disponemos de `Ctrl+c` para interrumpir el programa.

11.7 Clientes graficos de IRC

Como veremos en la próxima entrega, existen otros clientes de IRC para X Window con un sistema de manejo más sencillo: soportando multiventana, autoraise (es decir, que cuando cambia el contenido de una ventana esta cambia a primer plano), colores, menúes para los comandos más habituales, etc. Los veremos en versiones X Window, de txto (aunque más avanzados que ircII, como BitchX) e incluso específicos para

KDE (kirc, kvirc), y nos harán la comunicación en el IRC mucho más sencilla, sin olvidar los comandos que hemos aprendido hoy, que complementarán todo lo que les falte a estos clientes (al disponer de unas opciones determinadas se nos limita el control a lo dispuesto por el programa, mientras que mediante los comandos IRC, comentados hoy, podremos suplir cualquier necesidad que no esté disponible en los menúes).

TkIRC no es un cliente de IRC en sí mismo, sino que utiliza como programa de IRC el famoso ircII, proporcionando un front-end gráfico (una especie de menúes encargados de interactuar con el propio ircII), facilitando enormemente las tareas más habituales en el IRC. El programa puede obtenerse desde la página Web de sus creadores, www.north.de/ atte/tkirc, y ocupa apenas unos 100Kb. Para su utilización es necesario tener instalado el propio programa ircII, y el intérprete de Tcl/Tk, que es el lenguaje con el que está programado este front-end para XWindow. Además es posible añadirle scripts (ficheros con nuevos comandos y opciones) en tcl, lo cual le quita muchas limitaciones al programa (siempre puede ser ampliado por el usuario).

Kirc y Ksirc son 2 clientes de IRC para el Window Manager KDE. Son bastante visuales y disponen de interesantes opciones. Pueden obtenerse versiones estables de los mismos en www.kde.org, en la página de aplicaciones de KDE.

KvIRC es uno de los clientes de IRC más visuales y sencillos de manejar de todos: soporta colores, nick-completion (relleno de los nombres de las personas a las que hablamos cuando sólo escribimos las letras iniciales de los mismos), imágenes de fondo, configuración por medio de menúes, acceso sencillo a los comandos básicos del IRC por medio de una barra de iconos, integración con KDE, capacidad de lanzar netscape (o cualquier otro navegador) pinchando sobre cualquier dirección que aparezca en la conversación, sencilla instalación, mantenimieno y configuración, y un aspecto muy amigable (bastante similar al programa Mirc de Windows en cuanto a funcionamiento). Puede obtenerse en www.kvirc.org.

cIRCus es uno de los más sencillos clientes de IRC de Linux. Puede obtenerse en diferentes formatos (tar.gz y rpm) en la página Web de sus creadores, http://www.nijenrode.nl/ ivo/circus_data/download.html. En tiene características multiventana (al estilo de mIRC para Windows), la utilización de colores distintos para nuestros comentarios y los del resto de participantes en el IRC, así como la posibilidad de activar autoraise en las ventanas que deseemos. Autoraise es una propiedad que hace que cuando el contenido de una ventana cambia, ésta pase al primer plano, y sepamos que quien está hablando con nosotros ha dicho una nueva frase. Además soporta correctamente todos los acentos y carácteres españoles, y permite copiar/pegar con el mouse, sin olvidar que puede usarse arriba y abajo en los cursores para ir adelante y atrás en el historial de órdenes, pudiendo repetir así cualquier frase dicha en el pasado. Por otra parte, cIRCus tiene un sencillo menú en la parte superior con botones que sustituyen a las opciones de conexión, desconexión, entrada en canales (join), dar op, quitar op, dejar un canal, o salir del programa, pudiendo ejecutar además los comandos de IRC aprendidos el mes pasado, con lo que se puede decir que este programa aporta sencillez y funcionalidad a la potencia de los comandos del IRC. Su configuración se realiza editando un fichero de texto llano en nuestro directorio HOME (`/home/nombre/.circusrc`), desde donde podemos cambiar cualquier aspecto del programa: fuentes, menúes o identidades (al final del fichero, secciones Nick, IRCNAME, notify y Server List, las cuales nos permitirán especificar nuestro nick, nombre del cliente, lista de notificaciones y lista de servidores de IRC por defecto).

Zircon (http://catless.ncl.ac.uk/Programs/Zircon/) es otro cliente de IRC basado en Tcl/Tk (aunque este no necesita ircII para funcionar sino que utiliza las propias facilidades de red de este lenguaje). Necesitaremos al menos la versión 8.0.3 de este intérprete para su instalación, que se realiza desde un script de Tcl/Tk llamado installZircon quien se encargará de la correcta ubicación del programa en nuestro disco duro, además de permitirnos la lectura de documentos de ayuda y FAQs sobre Zircon.

Existen multitud de clientes de IRC gráficos y de texto para Linux. Una manera de ir directamente a un listado de clientes para poder seleccionar el que más nos interese es mediante www.freshmeat.net, tecleando irc en la celdilla de texto de búsqueda y pulsando Intro. Freshmeat nos mostrará toda la lista de programas relacionados con el IRC (muchos de ellos clientes) que haya en su base de datos, así como las direcciones de download y páginas de los mismos. Otro repositorio de clientes IRC para Linux está disponible en www.linuxberg.com, en la sección Software.

11.8 El cliente de IRC, BitchX

Uno de los clientes de IRC más potentes de los comentados aquí es BitchX, que trabaja en consolas de texto pero con facilidades propias de los clientes gráficos. Incluye gran cantidad de comandos y scripts, de tal modo que los más usados scripts de ircII han sido incluidos en este cliente como comandos del mismo. Es posible obtenerlo de ftp://ftp.bitchx.org (aunque también hay disponibles versiones rpm en el ftp de redhat). Asimismo, podemos obtener scripts y opciones adicionales para BitchX en http://www.phantasy.com/ markm/bitchx y http://scripts.bitchx.com. Está disponible tanto en rpm como en deb, tgz y código fuente, siendo éste último el más recomendable si deseamos habilitar el soporte para la fuente con caracteres en castellano o LATIN1.

Aunque por su nombre y la X final del mismo pudiera parecer que es un programa para XWindow, en realidad BitchX funciona en consolas de tipo texto o xterm, con lo que no necesitaremos el entorno de ventanas X para hacerlo funcionar (lo cual lo hace ideal para muchos usuarios). BitchX incluye una serie de características que lo hacen generalmente apreciado como cliente de IRC, gracias a opciones como NickCompletion, inclusión de los scripts más utilizados como comandos estándar del mismo, soporte multiventana (tanto partida como en ventanas virtuales), soporte de colores para distinguir los textos dirigidos a nosotros del resto de frases, modo historia de todo lo escrito (es decir, que se puede recorrer todo lo escrito mediante las teclas del cursor arriba y abajo, pudiendo reentrar cualquier frase al canal), tratamiento de la ventana como un buffer (con lo cual se puede subir y bajar por la misma con `AvPág` y `RePág`), etc.

Otra de sus principales características es que cuando alguien trata de floodearnos (es decir, enviarnos mucho texto muy rápidamente, tal vez tratando de molestarnos o estropear nuestra conexión), BitchX lo tira automáticamente de la conversación, además de incluir opciones de autoentrada en canales si alguien nos tira del mismo.

Asi pues, debe quedar claro que BitchX es uno de los clientes de IRC más completos y funcionales, con muchos seguidores (aunque también hay gente que no le gusta). Para iniciar BitchX se hace, pues, desde la línea de comandos, indicándole en esta el nickname que deseemos usar en el IRC, el servidor de IRC y el puerto (esto último opcional), todo ello en formato BitchX Nick Server:Puerto. Ejemplo:

```
[sromero@compiler BitchX]$ BitchX  Juan  irc.arrakis.es
```

Si no queremos especificar esto desde la línea de comandos es posible utilizar variables de entorno que definan el Nick, Servidor, y nombre del usuario para poder invocar al cliente de IRC simplemente mediante BitchX. Para ello BitchX proporciona al usuario la posibilidad de incluir estos datos en las variables de entorno IRCNAME, IRCNICK e IRCSERVER, tanto en ficheros de inicialización (.bashrc, .profile) como en la línea de comandos, tal y como puede verse en el siguiente ejemplo:

```
[sromero@compiler BitchX]$  export IRCNAME="Juan Sanchis"
[sromero@compiler BitchX]$  export IRCNICK="JuanChis"
[sromero@compiler BitchX]$  export IRCSERVER="irc.arrakis.es"
```

Además hay otros parámetros que podemos consultar en la página man de BitchX, como la autoentrada en canales:

```
BitchX -c \#linux
```

11.9 Como obtener el teclado en castellano en BitchX

Si el BitchX que estamos usando no dispone de soporte para caracteres latinos (acentos, eñes, etc.), podemos forzar a BitchX a que los visualice correctamente editando el fichero config.h de los fuentes del programa, habilitando (como pone en el README del programa) la opción para LATIN1, y recompilando el programa.

11.10 Comandos de BitchX

Con BitchX es perfectamente posible utilizar los comandos básicos del IRC vistos el mes pasado mediante ircII, es decir, /server, /quit, /join, /part, /query, etc.

Aparte de estos, ahora sí que es posible la utilización del comando /list para listar los canales del servidor de IRC. Recordemos que este comando también estaba disponible en ircII, pero la salida por pantalla era tan rápida que era imposible ver los canales disponibles. En BitchX la salida es igual de rápida, pero disponemos de las teclas AvPág y RePág para subir y bajar en el buffer de la ventana, y por tanto ver los canales una vez listados (o leer textos anteriormente escritos en un canal, consultar algo que dijimos, etc.).

Una vez se entra a un canal (con /join #canal), es posible ver la lista de operadores del canal mediante la tecla F2, y la lista de usuarios mediante F3 (aunque sigue pudiendo usarse el comando /names #canal).

Por último, y en cuanto a texto se refiere, cabe nombrar el soporte de colores y estilos de BitchX. Por ejemplo, para escribir un texto en negrita, basta con pulsar Ctrl+b (de bold, resaltado), y despues escribir el texto en cuestión, que aparecerá en negrita (ojo con esto pues los usuarios de ircII no lo verán así), y para escribir frases en colores, en cualquier punto de la frase puede utilizarse Ctrl+C y a continuación introducir en la frase el número de color (soltando Ctrl+C), con lo que aparecerá en el color correspondiente. Ejemplo:

```
> Hola a <CTRl+C>2todos!
```

Hará que aparezca "todos"en azul. Otros comandos que pueden. utilizarse en BitchX son /shell (para ejecutar copias de nuestra shell de comandos) /telnet (para hacer telnet a cualquier servidor desde dentro de BitchX), y /ftp (para hacer ftp desde el IRC), comandos no habituales en los restantes clientes de IRC

En BitchX también es posible la transferencia de ficheros vía DCC, para ello se disponen del típico comando /dcc send "nick fichero", del que ya disponíamos en ircII. La novedad de BitchX es que incluye por defecto la opción de aceptar todos los ficheros que te sean enviados, opción que se puede cambiar mediante /dcc auto off, de tal modo que cuando se nos envie un fichero, en lugar de ser aceptado automáticamente se nos informará de esto y mediante la tecla F5 se podrá obtener la lista de ficheros que nos son enviados, para aceptar cualquiera de ellos si nos interesa.

Otro comando importante es /set, que nos permitirá cambiar parámetros de configuración de BitchX, como por ejemplo desactivar el pitido que suena cuando nos hacen un Query en otra ventana (/set beep off). Más información mediante /help set o en la documentación del programa.

11.11 Nick Completion en BitchX

Una de las cosas más molestas en el IRC al haber diferentes personas en un canal y querer referirnos a una de ellas suele ser escribir el nombre del mismo, pues la mayoría de nombres en el IRC contienen carácteres extraños, son combinaciones de mayúsculas y minúsculas, o simplemente, puede apetecernos no escribir dicho nombre pero que esa persona sepa que le hablamos a ella. Para ello BitchX incorpora una funcionalidad conocida como Nick Completion, o completado de nicks, que consiste en que al pulsa la combinación de teclas CTRL+R, BitchX tratará de completar el nick que hayamos comenzado a escribir en la línea de comandos, esta tecla puede usarse en cualquier momento de la frase, y sus efectos se pueden observar en el siguiente ejemplo. Supongamos que existe un usuario en el IRC llamado AvIaToR, al cual queremos contestar o comentarle algo:

```
> Av<CTRL+R>: Como estas?
> Ya te lo he dicho antes, Av<CTRL+R>...
```

Estas 2 frases se expanderán automáticamente a:

```
> AvIaToR: Como estas?
> Ya te lo he dicho antes, AvIaToR...
```

Otra posibilidad es utilizar la combinación de teclas Control más la tecla de '+' (`Ctrl+'+'`), que al principio de una frase completa el nick que hayamos comenzado a escribir y lo completa con 2 puntos (:), para dirigir frases directamente a cualquier usuario:

```
> Av<CTRL+'+'> lo estoy buscando...
```

Se expande a:

```
> AvIaToR: lo estoy buscando...
```

Una última posibilidad es la utilización de la tecla TAB al principio de una frase, que se expanderá a un /msg Nick, donde Nick será uno de los usuarios del canal y cambiará dependiendo del número de veces que sea pulsada (una vez, aparece el primer usuario, otra vez, el segundo, etc.). Esto nos permitirá enviar mensajes con relativa facilidad a cualquier usuario del canal.

11.12 BitchX: Soporte multiventana

Una de las características principales de BitchX es el soporte multiventana nativo. Al igual que en ircII, es posible entrar en varios canales o conversaciones simultáneamente, y que aparezcan todos ellos en la ventana principal de BitchX, con una indicación al principio de cada uno de ellos que nos indique de qué canal viene cada frase. Esto es muy molesto y bastante inviable, y además solo hay un canal activo (al que va el texto que escribimos) que podemos cambiar mediante la tecla CTRL+X (con lo que irá cambiando entre los diferentes canales a cada pulsación de la misma).

Para evitar esto BitchX incluye soporte multiventana tanto en pantalla partida como en ventanas virtuales, es decir, es posible partir la pantalla en diferentes secciones y disponer de un canal o un query en cada una de ellas, así como crear diferentes pantallas virtuales (de la que sólo vemos una, la visible) y cambiar entre ellas, con lo que dispondríamos de una ventana aislada para cada canal, y el texto no se mezclaría.

Para partir la pantalla es posible utilizar el comando /window new, y para crear una pantalla virtual (mucho más recomendable) se utiliza el comando /window new hide. Este comando puede ser utilizado tantas veces como pantallas virtuales se deseen, de tal modo que irán siendo numeradas desde el 2 (la 1 es la pantalla que aparece por defecto) hasta n (donde n es el número de pantallas creadas). Una vez se han creado las diferentes ventanas es posible cambiar entre ellas mediante ALT+nº (ALT+1, ALT+2, etc.) en la consola de texto o mediante ESC-nº si estamos en XWindow (en este último caso, no consiste en pulsar ambas teclas simultáneamente sino en pulsar ESC, liberar la tecla y pulsar un número de la fila superior de números, no del teclado numérico).

De esta forma, si deseamos disponer de 3 ventanas virtuales, con el canal #linux en la primera, #programacion en la segunda, y una conversación privada en la tercera, es posible hacer lo siguiente:

```
> /window new hide
> /window new hide
> /join #linux
> ALT+2 o ESC-2
```

```
> /join #programacion
> ALT+3 o ESC-3
> /query JuanChis
```

Una vez hecho esto dispondremos de las 3 ventanas creadas y ya podremos ir a cualquiera de ellas a decir las frases relativas a cada canal, sin que las salidas se mezclen en pantalla ya que BitchX nos separará la salida de las mismas a cada canal. Se puede decir, pues, que cada comando /join, /part o /query se aplica a la pantalla virtual que tengamos activa en ese momento. A continuación se ofrece un extracto de la ayuda en línea proporcionada por BitchX para el uso de combinaciones de teclas y la diferentes opciones relativas a ventanas (tanto secciones de pantallas partidas como ventanas virtuales):

```
> Control+W+'-' = Hace la sección actual una línea más estrecha.
> Control+W+'+' = Hace la sección actual una línea más ancha.
> Control+W+l     = Lista las ventanas disponibles.
> Control+W+k    = Elimina la ventana/sección actual.
> Control+W+h    = Oculta la ventana/sección actual.
> Control+W+b    = Iguala el tamaño de las secciones.
> Control+W+n    = Cambia a la ventana siguiente.
> Control+W+p    = Cambia a la ventana anterior.
```

Aparte de estas teclas, están disponibles AvPág y RePág para moverse por el buffer de cualquier ventana y los cursores arriba y abajo para recuperar frases anteriormente escritas.

11.13 Como utilizar scripts en BitchX

Para utilizar scripts en BitchX (funciones y comandos nuevos para un cliente de IRC), tan sólo hay que copiarlos en el directorio .BitchX y cargarlos luego en el programa mediante /load script, teniendo el comando disponible a partir de ese momento. Lo importante de BitchX es que los scripts pueden proporcionar un aspecto diferente al entorno del programa, al ser este totalmente programable, con lo que las posibilidades de BitchX son realmente ilimitadas.

Los scripts se pueden obtener a partir de la página Web del propio programa, http://www.bitchx.com o en http://scripts.bitchx.com.

11.14 Instalar un servidor de IRC

Mediante Linux no sólo podemos conectarnos a los IRC existentes (como el IRC Hispano, por ejemplo) utilizando clientes de irc, sino que también es posible obtener *servidores* de IRC para convertir nuestra máquina en punto de encuentro de amiguetes o usuarios varios. Para ello simplemente hay que obtener, instalar y configurar un servidor de IRC en nuestra máquina, como ircd. La versión que utiliza el autor es la 10.02, (fichero ircu2.10.02-LinuxELF.tar.gz, ver http://www.argo.es/ jcea/irc/ircd.html, una estupenda página sobre redes e IRC), que consiste en un fichero tar.gz descomprimible con «tar -xvzf ircu2.10.02-LinuxELF.tar.gz», con los siguientes ficheros:

```
[sromero@compiler ircd]$ ls
Authors       LICENCE       config.h      ircd*
ircd.motd     ircd.rehash*  INDEX         README
example.conf  ircd.conf     ircd.off*     remote.motd
```

(pueden cambiar según versiones, y también está disponible en formato deb y rpm en los sites correspondientes).

Una vez descomprimido (aunque también están disponibles las fuentes para compilar el programa adaptándolo a nuestras necesidades), y como se comenta en el README del programa, basta con editar el fichero de configuración ircd.conf y adaptarlo a las funcionalidades necesarias, para luego ejecutar el daemon ircd. Esto se hace mediante la ejecución del programa con el comando./ircd, y como un usuario que no sea root, preferentemente llamado irc) para realizar la inicialización del daemon, y así permitir hasta 252 conexiones a nuestro sistema. Lo más recomendable es instalar la versión rpm (o deb, conviertiendola mediante alien) creando previamente un usuario llamado irc (con adduser) y cambiando su password a nada (a una cadena vacía, editando `/etc/passwd` o mediante el comando passwd), haciendo lo siguiente:

```
[sromero@compiler ircd]$  adduser irc
[sromero@compiler ircd]$  rpm -i  ircd.rpm
```

Para finalizar el uso del daemon de irc basta con ejecutar el script ircd.off. Este rpm (creado por Pablo Saratxaga, habitual de es.comp.os.linux) puede obtenerse de la Web de S.u.S.E (www.suse.com), entrando en su FTP, o desde cualquiera de sus mirrors: ftp://ftp.tu-chemnitz.de/pub/linux/suse/5.3/i386.de/suse/n1/ (fichero ircd.rpm), así como en los FTPs de las otras distribuciones Linux (Redhat, Debian, etc.). La versión empaquetada por Pablo Saratxaga dispone además del fichero ircd.conf correctamente configurado para el servidor de IRC, y tan sólo habremos de modicarlo si queremos que ircd corra en una red de servidores. Otro fichero interesante es ircd.motd, que nos permitirá variar el mensaje a la entrada de nuestro IRC. Una vez instalado el daemon/servidor de irc, es posible testear nuestro servidor accediendo a nuestra propia máquina:

```
[sromero@compiler ircd]$  irc  Juan  localhost:6667
```

Si todo está correctamente instalado será posible entrar en diferentes consolas virtuales y repetir el proceso para tener varios usuarios en nuestro IRC. Ya sólo nos falta, pues, editar el fichero ircd.conf para dotar a nuestro irc de los canales, comandos y datos que tengamos pensados para él, y dar nuestra dirección IP a las personas que queramos que se conecten al mismo.

12 Comandos basicos

12.1 ¿Pero hay que saber comandos?

La respuesta a esta pregunta es sencilla, todo depende de como vayas a utilizar el sistema, de las ganas de aprender que tengas y de lo que quieras profundizar en el sistema.

Hay que diferenciar entre *tareas de administracion* y *uso habitual del sistema*.

Si vas a realizar tareas de administracion del sistema, deberias saber, o mejor dicho, tendras que aprender una serie de comandos para realizar estas tareas. Si vas a utilizar el sistema a nivel usuario, no necesitaras saber ningun comando, solamente tendras que aprender a utilizar los programas que uses.

Lo que ocurre habitualmente es que el usuario de la maquina y el administrador son la misma persona, con lo cual tendran que saber una serie de comandos minimos.

Como anecdota, os puedo contar que en lugar donde trabajo, hay usuarios que utilizan Linux y no tienen ni idea de administracion del sistema (y no solo en linux), no saben comandos y tampoco los necesitan. Trabajan en modo grafico, saben utilizar los programas que necesitan y tienen a alguien (yo en este caso ;-)) que se hace cargo de las tareas de administracion/configuracion del sistema.

12.2 ¿Esto no lo puedo hacer en modo grafico?

Como regla general, se podria decir lo siguiente: *"Todo lo que se puede hacer en modo grafico, se puede hacer tambien en modo texto, a base de comandos. Pero no todo lo que se puede hacer en modo texto, se puede* *hacer en modo grafico*".

Cada dia aparecen mas y mas programas, que permiten hacer hacer en modo grafico, a golpe de raton y ventanas, lo que se hace en modo texto a base de comandos. Es mas, lo unico que estos programas hacen, es crear una version grafica que hace uso de los comandos que se utilizan en modo texto. Entornos graficos como Gnome y Kde, vienen con una serie de programas graficos que hacen esto.

Para terminar esta seccion, solo decir a aquellos que quieran profundizar en el sistema, que hasta el momento, ningun programa grafico, supera en versatilidad y potencia, a una consola de texto donde ejecutar comandos directamente.

12.3 Combinaciones especiales de teclas

- `[Ctrl]+[Alt]+[Del]`: Shutdown. Apaga el sistema de forma organizada desde una terminal texto.
- `[Ctrl]+[Alt]+[F1]`: Cambia a la primera terminal de texto.
- `[Ctrl]+[Alt]+[Fn]`: Cambia a la terminal de texto numero `n (n=1,...,6)`
- `[Ctrl]+[Alt]+[F7]`: Cambia a la primera terminal X (si se esta usando alguna)
  NOTA:Si estamos en X-windows y por alguna razon nuestro entorno grafico dejase de funcionar, siempre podriamos entrar en una terminal texto con `[Ctrl]+[Alt]+[Fn]`, terminar los procesos que esten dando problemas y volver a las X con `[Ctrl]+[Alt]+[F7]`.
- `[Ctrl]+[Alt]+[Fn]`: Cambia a la terminal X numero `n (n=7,...,11)`
- `[Tab]`: Autocompleta el nombre de un comando, fichero, directorio, programa, cuando trabajamos en una terminal texto. P.ej: Si vamos a usar el programa "programa_de_prueba", podemos escribir progr y pulsar `[Tab]`, el sistema se encargara de rellenar el resto.
- `[ArrowUp]`: (Flecha arriba) Va editando la historia de comandos que hemos escrito anteriormente en terminal texto.`[Enter]` para ejecutar.
- `[Shift][PgUp]`: Scroll la salida de la terminal hacia arriba, en terminal texto.
- `[Shift][PgDown`: Scroll la salida de la terminal hacia abajo, en terminal texto.
- `[Ctrl]+c`: Termina el proceso actual. Cuando lo comenzamos sin & (`[localhost]$ proceso`)
- `[Ctrl]+d`: Termina la terminal actual.
- `[Ctrl]+s`: Para la transferencia a la terminal.
- `[Ctrl]+z`: Manda el proceso actual (comenzado sin &) a "Background". Lo mismo que si comenzamos el proceso con & (`[localhost]$ proceso &`).

12.4 Lista de comandos mas usados

Todos los comandos citados en este apartado, pueden usarse con multitud de opciones. Para conseguir informacion detallada de cada comando utilizar `man` *`comando`*. A continuacion teneis la manera mas sencilla de utilizacion.

Informacion del sistema

- `hostname`: Devuelve el nombre de la maquina.
- `uptime`: Devuelve la cantidad de tiempo trascurrido desde la ultima vez que se arranco el sistema, la cantidad de usuarios trabajando en el sistema y el load average (carga del sistema).

- `uname -a`: Informacion sobre el sistema operativo de la maquina.
- `dmesg|more`: Imprime el "ring buffer"del kernel.
- `free -tm`: Informacion sobre la cantidad de memoria disponible y usada.
- `df -h`: Informacion sobre todo los dispositivos montados en la maquina.
- `du -bh /|more`: Informacion sobre el espacio ocupado por cada subdirectorio, comenzando en el directorio raiz /.
- `ps`: Informacion sobre los procesos del actual usuario, que se estan ejecutando.
- `ps axu`: Informacion sobre todos los procesos que se estan ejecutando en la maquina.
- `top`: Informacion sobre el uso de cpu y memoria de los procesos del sistema.
- `cat /proc/cpuinfo`: Informacion sobre el microprocesador.
- `cat /proc/interrupts`: Informacion sobre las interrupciones en uso.
- `cat /proc/dma`: Informacion sobre dma en uso.
- `cat /proc/filesystems`: Informacion sobre los sistemas de archivos que se pueden utilizar (compilados en el kernel).
- `/sbin/lsmod`: Informacion sobre los modulos en uso.
- `finger` *`usuario`*: Informacion sobre el usuario *usuario*.
- `who`: Informacion sobre los usuarios usando el sistema.
- `id` *`usuario`*: Informacion sobre UID, GID y GROUPS del usuario *usuario*
- `last`: Informacion sobre los ultimos usuarios que han usado el sistema.
- `set|more`: Informacion sobre el entorno de usuario actual.
- `ping` *`maquina`*: Para comprobar si tenemos contacto con la maquina *maquina* (*maquina*: nombre o IP)
- `/sbin/route`: Tabla de enrutamiento de nuestro sistema.
- `/sbin/ifconfig`: Informacion sobre los distintos dispositivos de red
- `netstat`: Informacion valiosa sobre la conexion de red (Este comando tiene muchas posibilidades, ejecutar `man netstat`

Administracion

- `su`: Te conviertes en administrador(root) despues de introducir la clave de acceso. Ideal para realizar alguna tarea de administracion sin necesidad de salir del sistema y entrar de nuevo como root.
- `/usr/sbin/adduser` *`usuario`*: Registra y crea una cuenta de usuario.
- `/usr/sbin/userdel` *`usuario`*: Borra la cuenta de usuario *usuario*.
- `passwd`: Cambia la clave de acceso para el usuario actual. Root puede cambiar la clave de cualquier usuario `passwd` *`usuario`*
- `/usr/sbin/groupadd` *`grupo`*: Crea un nuevo grupo.

- `/usr/sbin/groupdel` *`grupo`*: Borra un grupo.
- `more /etc/passwd`: Muestra el fichero de claves del sistema. (Si no se usa shadow password)
- `more /etc/group`: Muestra los grupos registrados en el sistema.
- `chmod` *`permisos fichero/directorio`*: Cambia los permisos de ficheros/directorios
- `cp` *`fichero1 fichero2`*: Copia fichero1 como fichero2
- `rm` *`fichero`*: Borra fichero
- `rm -R` *`directorio`*: Borra el contenido completo (ficheros/subdirectorios) de directorio
- `mv` *`fichero1 fichero2`*: Cambia el nombre de fichero1 a fichero2
- `mkdir` *`directorio`*: Crea un subdirectorio
- `rmdir` *`directorio`*: Borra un subdirectorio
- `shutdown -t3 -r now`: Reinizializa el sistema (hay que hacerlo como root).
- `shutdown -t3 -h now`: Apaga el sistema (hay que hacerlo como root).

13 Servidores FTP

Aqui teneis una seleccion de servidores FTP, en donde es posible encontrar casi todo lo referente a Linux, programas, aplicaciones, distribuciones, documentacion, ...

Europa

- ***ftp://ftp.univie.ac.at/systems/linux/*** `<ftp://ftp.univie.ac.at/systems/linux/>`
- ***ftp://ftp.funet.fi/pub/Linux*** `<ftp://ftp.funet.fi/pub/Linux>`
- ***ftp://ftp.univ-angers.fr/pub/Linux*** `<ftp://ftp.univ-angers.fr/pub/Linux>`
- ***ftp://ftp.gwdg.de/pub/linux/*** `<ftp://ftp.gwdg.de/pub/linux/>`
- ***ftp://2cows.netsurf.de/pub*** `<ftp://2cows.netsurf.de/pub>`
- ***ftp://ftp.uni-paderborn.de/pub/Mirrors/sunsite.unc.edu*** `<ftp://ftp.uni-paderborn.de/pub/Mirrors/sunsite.unc.edu>`
- ***ftp://ftp.rz.uni-ulm.de/pub/systems/linux/*** `<ftp://ftp.rz.uni-ulm.de/pub/systems/linux/>`
- ***ftp://ftp.dfv.rwth-aachen.de/pub/linux*** `<ftp://ftp.dfv.rwth-aachen.de/pub/linux>`
- ***ftp://ftp.tu-dresden.de/Linux*** `<ftp://ftp.tu-dresden.de/Linux>`
- ***ftp://cnuce-arch.cnr.it/pub/Linux*** `<ftp://cnuce-arch.cnr.it/pub/Linux>`
- ***ftp://src.doc.ic.ac.uk/packages/linux/sunsite.unc-mirror*** `<ftp://src.doc.ic.ac.uk/packages/linux/sunsite.unc-mirror>`
- ***ftp://unix.hensa.ac.uk/mirrors/sunsite/pub/Linux*** `<ftp://unix.hensa.ac.uk/mirrors/sunsite/pub/Linux>`
- ***ftp://ftp.nvg.unit.no/pub/linux*** `<ftp://ftp.nvg.unit.no/pub/linux>`

- ***ftp://ftp.kfki.hu/pub/linux*** `<ftp://ftp.kfki.hu/pub/linux>`
- ***ftp://ftp.fi.muni.cz/pub/UNIX/linux*** `<ftp://ftp.fi.muni.cz/pub/UNIX/linux>`
- ***ftp://ftp.etsimo.uniovi.es/pub/linux*** `<ftp://ftp.etsimo.uniovi.es/pub/linux>`
- ***ftp://sunsite.cnlab-switch.ch/mirror/linux*** `<ftp://sunsite.cnlab-switch.ch/mirror/linux>`
- ***ftp://ftp.informatik.rwth-aachen.de/pub/gnu*** `<ftp://ftp.informatik.rwth-aachen.de/pub/gnu>`

America

- ***ftp://metalab.unc.edu/pub/Linux/*** `<ftp://metalab.unc.edu/pub/Linux/>`
- ***ftp://tsx-11.mit.edu/pub/linux*** `<ftp://tsx-11.mit.edu/pub/linux>`
- ***ftp://ftp.cc.gatech.edu/pub/linux*** `<ftp://ftp.cc.gatech.edu/pub/linux>`
- ***ftp://prep.ai.mit.edu/pub/gnu*** `<ftp://prep.ai.mit.edu/pub/gnu>`

Asia

- ***ftp://ftp.spin.ad.jp/pub/Linux*** `<ftp://ftp.spin.ad.jp/pub/Linux>`

Africa

- ***ftp://ftp.is.co.za/linux*** `<ftp://ftp.is.co.za/linux>`

Otros

- ***ftp://ftp.cdrom.com/pub/linux*** `<ftp://ftp.cdrom.com/pub/linux>`
- ***ftp://ftp.linux.org/pub/mirrors*** `<ftp://ftp.linux.org/pub/mirrors>`
- ***ftp://ftp.rge.com/pub/systems/linux*** `<ftp://ftp.rge.com/pub/systems/linux>`
- ***ftp://ftp.nuri.net/pub/Linux*** `<ftp://ftp.nuri.net/pub/Linux>`
- ***ftp://ftp.nus.sg/pub/unix/Linux*** `<ftp://ftp.nus.sg/pub/unix/Linux>`
- ***ftp://ftp.nectec.or.th/pub/mirrors/linux*** `<ftp://ftp.nectec.or.th/pub/mirrors/linux>`
- ***ftp://ftp.cs.cuhk.hk/pub/Linux*** `<ftp://ftp.cs.cuhk.hk/pub/Linux>`

14 ¿Problemas?

14.1 ¿Quien me puede ayudar a resolver un problema?

Lo mejor que se puede hacer para encontrar solucion a un problema con linux es leer, leer la documentacion existente.En la siguiente subseccion teneis informacion sobre donde encontrar documentacion.

Si despues de leer la documentacion, sigues sin poder solucionar tu problema, te aconsejo que mandes un mensaje a alguna lista de correos o a un grupo de noticias, que trate sobre linux.

- Informacion sobre diferentes listas de correo sobre linux, se puede encontrar en *http://linux-es.uio.no/listas.cgi* `<http://linux-es.uio.no/listas.cgi>`.
- Informacion sobre diferentes grupos de noticias sobre linux se puede encontrar en *http://linux-es.uio.no/news.cgi* `<http://linux-es.uio.no/news.cgi>`. El grupo de noticias en castellano `es.comp.os.linux` es uno de los mejores sitios para conseguir ayuda.

Como nota adicional decir que, si has comprado una distribucion de Linux, existen varias (RedHat, SuSE, OpenLinux, ...) que dan soporte de instalacion por un periodo de tiempo determinado.

14.2 ¿Dónde encuentro documentacion sobre linux?

Existe un proyecto de documentacion de Linux (LDP), que se encarga de coordinar/distribuir todos los documentos existentes sobre linux y temas realcionados, los famosos manuales, Howtos y Mini-Howtos.

- **En castellano** tenemos a *Lucas* `<http://lucas.hispalinux.es/>` e *Insflug* `<http://www.insflug.org/>`, dos proyectos que se dedican a la creacion de documentacion en castellano y a la traduccion de los documentos en ingles del LDP.
- **En ingles** tenemos al *Linux Documentation Project (LDP)* `<http://metalab.unc.edu/LDP/>`.

15 Colaboradores

Ante todo dar las gracias a todos los colaboradores que han participado en la creacion de este documento. Sin su ayuda, sin sus conocimientos y sin el tiempo que han dedicado, no hubiese sido posible crearlo. A continuacion teneis una lista de las personas que han participado activamente en la redaccion/creacion de la FAQ sobre linux en castellano.

- Rafael Martinez, r.m.guerrero@medisin.uio.no, Universidad de Oslo
  `Coordinador.`
  `Secciones 1, 5, 12, 13, 14, 15, 16, 17`
  `Subsecciones 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 6.1, 6.7, 6.9, 7.11`
- A. Gustavo Gonzalez, agonzale@cica.es, Universidad de Sevilla
  `Colaborador`
  `Seccion 4`
  `Subsecciones 2.1, 2.9, 8.6`
- Mario Teijeiro Otero, asimovi@teleline.es
  `Colaborador`
  `Subseccion 6.3, 6.4, 6.5, 7.1, 7.2, 7.3, 7.5, 7.12`
- Manuel Constantino, constan@bart.us.es
  `Colaborador`
  `Subsecciones 8.1, 8.2, 8.3, 8.4, 8.5, 8.7, 8.8, 8.9, 8.10, 8.11, 8.12, 8.13, 8.14.`
- Santiago Romero Iglesias, santiago.romero@iname.com
  `Colaborador`
  `Secciones 9, 10, 11`
  `Subsecciones 3.9, 6.2, 6.6, 6.8, 7.6, 7.7, 7.8, 7.9, 7.10`

16 Por Hacer

Aqui teneis una lista con las secciones e ideas que tenemos pensado incluir en futuras versiones de este documento:

- Revision de lo ya publicado, despues de recibir comentarios de los usuarios.
- Seccion Trucos y ajustes del sistema ¿?
- Seccion diccionario de bolsillo ¿?
- Subseccion ¿Qué paquetes instalo?
- Subseccion ¿Qué entorno grafico es el mejor?
- Subseccion ¿Cómo instalo un nuevo programa?
- Subseccion ¿Cómo configuro la impresora?
- Subseccion ¿Cómo instalo mi ZIP-drive?

17 Copyright