

# LINUX

## Preparación para la certificación LPIC-1 (exámenes LPI 101 y LPI 102) - 2<sup>a</sup> edición

Los exámenes **LPI 101 y LPI 102** son los dos exámenes que permiten obtener la **certificación LPIC-1** "Junior Level Linux Professional". Este programa de certificación del Linux Professional Institute es cada vez más **reconocido por los responsables de selección**, que ven en esta certificación un requisito previo a la contratación o al acceso a un puesto de administrador.

Los exámenes LPI 101 y 102 demuestran a los profesionales que usted domina las bases de la administración de **Linux, sea cual sea su distribución**: la instalación y la configuración completa de un **puesto de trabajo o de un servidor** así como todos los servicios asociados, tanto **sistemas** como **redes**. Esta nueva edición del libro cuenta con las **últimas novedades de Linux**(núcleo 3.0, IPv6, nuevas distribuciones, nuevo bootloader, etc.).

Para ayudarle a preparar de manera eficaz esta certificación, el libro abarca**todos los objetivos oficiales de la última versión del examen**, tanto desde un punto de vista teórico como práctico. Está redactado por un formador profesional reconocido, también consultor, certificado en Linux. Así, los conocimientos pedagógicos y técnicos del autor conducen a un enfoque claro y visual, de muy alto nivel técnico.

Capítulo a capítulo, podrá **validar sus conocimientos teóricos**, gracias a las múltiples **preguntas y respuestas (536 en total)** que ponen de relieve tanto los elementos fundamentales como las características específicas a los conceptos tratados.

Cada capítulo finaliza con unas **prácticas (47 en total)**, con las que tendrá los medios para medir su autonomía. Estos tests concretos, que van incluso más allá de los objetivos fijados por el examen, le permitirán forjarse una primera experiencia significativa y adquirir verdaderas competencias técnicas en situaciones reales.

Además del dominio del producto y de los conceptos, se añade la preparación específica para la certificación: podrá acceder de **manera gratuita a un examen de prueba en línea**, destinado a entrenarle en condiciones similares a las de la prueba. En la web, cada pregunta que se plantea se inscribe en la idea de la certificación y las respuestas se comentan de manera suficiente para identificar sus últimas lagunas. ¡Juzgue usted mismo si está listo para el examen!

### Los capítulos del libro:

Descripción – Introducción – Presentación de Linux – Instalación de Linux y de los paquetes de software – El shell y los comandos GNU – Los discos y el sistema de ficheros – Inicio de Linux, servicios, núcleo y periféricos – Las tareas administrativas – La red – La seguridad – X Window – Particionamiento avanzado: RAID y LVM – Anexo – Lista de objetivos

---

### Sébastien ROHAUT

**Sébastien Rohaut** es Ingeniero de Sistemas en proyectos para grandes clientes. También enseña Unix y PHP en clases preparatorias y de ingenieros, y prepara a los estudiantes para las certificaciones LPIC. Él mismo se ha certificado Linux y está muy comprometido con el mundo de los programas libres (fundador y antiguo presidente de Slyunix, asociación de promoción de Linux). Escribe con frecuencia en prensa especializada (Planeta Linux...) artículos destinados a aficionados de Linux y a la programación libre.

## Descripción

Los exámenes **LPI 101 y LPI 102** permiten obtener la **certificación LPIC-1** Junior Level Linux Professional. Este programa de certificación del Linux Professional Institute goza cada vez más **del reconocimiento por parte de los reclutadores**, que ven en esta certificación un requisito para la contratación o el acceso a un puesto de administrador.

Los exámenes LPI 101 y 102 demuestran a los profesionales que usted domina las bases de la administración de un sistema **Linux sea cual sea la distribución**: la instalación y configuración completa de un puesto de trabajo y de todos los servicios asociados a él, tanto de sistemas como de redes.

Para ayudarle a preparar de manera eficiente esta certificación, este libro abarca los objetivos oficiales cuya lista se encuentra en el anexo. Se divide en diez capítulos que comportan la siguiente organización:

- Una definición de los objetivos que se deben alcanzar: permite exponer de manera precisa las competencias proporcionadas por el capítulo una vez validado este último.
- Una parte de **cursos teóricos**: permite definir los términos y conceptos tratados y esquematizar en la forma de hilo conductor los diferentes puntos que hay que asimilar.
- Una parte de **aplicación del curso**: permite seguir el desarrollo preciso del tratamiento del tema con imágenes de pantalla y esquemas.
- Una parte de **validación de los conocimientos** propuesta en forma de preguntas y respuestas (516 en total). Estas preguntas ponen de relieve tanto los elementos fundamentales como las características específicas de los conceptos tratados. La parte **respuestas** retoma las preguntas planteadas con respuestas redactadas para cada una de ellas.
- Las **prácticas**: permiten ilustrar algunas partes del curso y también le proporciona los medios de medir sus conocimientos. Estas prácticas concretas le permitirán, incluso más allá de los objetivos fijados por el examen, forjarse una primera experiencia significativa y adquirir verdaderas competencias técnicas mediante situaciones reales.

Para la preparación específica al examen, puede acceder **de manera gratuita a un examen de prueba** en línea en la dirección <http://www.edieni.com/espanol/certificaciones> y, así, controlar sus conocimientos en condiciones cercanas a las de la prueba real. En este sitio Web, cada pregunta hecha se inscribe en el marco de la certificación, y cada respuesta se comenta de manera suficiente, lo que le permitirá identificar y resolver sus últimas lagunas.

## ¿Para qué certificarse?

Si busca empleo en informática y, en particular, en el desarrollo, explotación, producción, administración e ingeniería de sistemas, Linux es para usted casi una parada obligada. Debe conocer Linux y tener, en consecuencia, un apartado sobre Linux en su currículum vitae.

Muchos de los lectores de este libro han aprendido Linux en casa o en una escuela. Algunos son apasionados y la mayoría se han formado de manera autodidacta, ya sea en su hogar o gracias a su experiencia en una empresa. Son un grupo de personas importante y la mayoría de las empresas se basan en esta experiencia para contratarlas.

Pero en ocasiones esto ya no basta. Es cada vez más corriente encontrar anuncios que piden una certificación en este o aquel nivel, resaltando que tal cosa sería una ventaja o, incluso, un requisito indispensable.

Linux se ha profesionalizado y, como en el caso de otros campos, la formación en la materia se ha hecho indispensable. Por esta razón algunos organismos proponen certificar mediante los niveles correspondientes la posesión de dichas competencias. El caso típico es el de un fabricante de software que le ofrece certificarse en su programa o sistema. Le entrega un título, un certificado que prueba la validez de sus conocimientos.

Han pasado dos años desde la publicación de la edición anterior. El estado del mercado laboral en el ámbito de las tecnologías de la información nos reafirma en nuestra opinión. Con Linux evolucionando y haciéndose más complejo, la formación básica o autodidacta quizá no es valor suficiente a los ojos de un responsable en la contratación de personal. Disponer de una certificación es una garantía de conocimientos y de aptitud de cara a ser rápidamente operativo para la empresa.

Las certificaciones no son diplomas y por lo tanto no tienen valor como tal, pero las empresas las reconocen y las aceptan como garantía de competencias profesionales.

La propia naturaleza de Linux y, concretamente, la multiplicidad de las distribuciones, hace que no sea nada sencillo proveer un programa de certificación común al conjunto. Por eso, algunos distribuidores como Red Hat y Novell proponen ciclos formativos que conducen a su certificación.

# Las certificaciones LPI

El **Linux Professional Institute, LPI**, es una organización sin ánimo de lucro (una NPO: *Non Profit Organization*) creada en diciembre de 1999 en Canadá. Su objetivo es promover las tecnologías Linux, Free and Open Source a través de la creación de programas y exámenes de certificación de gran calidad. Los programas y exámenes de las certificaciones son **independientes**: no se ciñen a una distribución particular (excepto si se trata de una certificación especializada, como la de Ubuntu por ejemplo), sino al conjunto de las técnicas comunes a todas las distribuciones Linux, así como a los puntos esenciales según los tipos de distribuciones (por ejemplo, tanto el uso de dkpg como de rpm).

Grandes grupos como IBM, Novell, SGI, NEC se han unido a la iniciativa del LPI, que, por otro lado, ha recibido el apoyo de varios editores de distribuciones y revistas especializadas.

En resumen, LPI ha desarrollado una serie de tres certificaciones:

- **LPIC-1**: nivel **Junior**, compuesto de dos exámenes (LPI 101 y LPI 102), que certifica que la persona puede:
  - Trabajar en línea con los comandos Linux; ejecutar las tareas de mantenimiento sencillas: ayuda a los usuarios, creación de cuentas de usuario, gestión de un sistema extendido, copias de seguridad y restauración, paro y reinicialización (reboot).
  - Instalar y configurar un puesto de trabajo (que tenga instalado el gestor de ventanas X Window) y conectarlo a una LAN (red local), o a un PC independiente, mediante un módem conectado a Internet.
- **LPIC-2**: nivel **Intermedio**, compuesto de dos exámenes (LPI 201 y LPI 202), que certifica que la persona puede:
  - Administrar una implementación de pequeña o mediana envergadura.
  - Diseñar, instalar, mantener, dotar de seguridad y arreglar una pequeña red: conexión a Internet (cortafuegos, proxy, mail), gestión de servidores de Internet (servidor Web, servidor FTP), trabajo en equipo.
- **LPIC-3**: nivel **Senior**, compuesto de tres exámenes (LPI 301, LPI 302 y LPI 303), que certifica que la persona puede:
  - Definir y administrar el acceso individualizado a varios servicios (LDAP, PAM y tecnologías asociadas).
  - Diseñar y administrar la integración multiservicios de una red mixta, incluso multisitios (Samba, NFS y tecnologías asociadas).
  - Gestionar la complejidad (automatización, seguridad, equipos, etc.).

Según el LPI, una certificación LPIC es:

- **Profesional**: establecida por una comunidad de profesionales de Linux, voluntarios, profesores y proveedores de servicios informáticos.
- **Un desafío**: si usted no se sabe todo el temario, no la obtendrá.
- **Accesible**: disponible en miles de centros examinadores en todo el mundo permanentemente y de forma temporal durante eventos importantes.
- **De gran calidad**: sigue los consejos y recomendaciones de expertos Linux reputados y emplea una metodología de examen de tipo psicométrico reconocida en ámbitos científicos e industriales.
- **Neutra**: no se apoya en ninguna distribución específica, sino en conocimientos que se aplican a cualquier sistema Linux estándar.

- **Pertinente:** se prueba y se elige las preguntas teniendo en cuenta cuáles son pertinentes en un entorno real.
- **Independiente** con respecto a organismos de formación: no hay una guía única o un curso único de formación. Se fomenta todo tipo de método de formación.
- **Compatible y patrocinada** por empresas muy grandes especializadas o no en Linux.
- **Un orgullo:** una certificación respetada en la comunidad Linux y de la cual se enorgullece.

# La certificación LPIC-1

## 1. Los objetivos

Este libro tiene como objetivo proporcionarle todas las herramientas precisas para que apruebe la certificación LPIC-1. Esta certificación requiere la superación de dos exámenes: el LPI 101 y el LPI 102. Usted debe aprobar ambos para estar certificado en LPIC-1. La última revisión del programa LPIC-1 se realizó en 2009. Generalmente, se realiza una revisión cada cuatro o cinco años.



La lista de objetivos que se encuentra al final de esta obra retoma todos los objetivos de cada uno de los exámenes y le indica dónde encontrar los puntos correspondientes en el libro, así como los ejercicios asociados.

A continuación se presentan los objetivos principales del examen LPI 101:

- 101: Arquitectura y hardware.
- 102: Instalación de Linux y gestión de los paquetes.
- 103: Los comandos GNU y Unix.
- 104: Periféricos, sistema de ficheros, estructura en árbol.

Si supera este primer examen, usted será capaz de instalar una distribución Linux clásica, a saber, trabajar con la línea de comandos y las instrucciones del shell, manejar la estructura en árbol (el sistema de ficheros y los directorios), los discos y las particiones, instalar programas complementarios y trabajar en un entorno gráfico. Por lo tanto, determina si usted puede utilizar de manera sencilla un puesto de trabajo Linux.

A continuación se presentan los objetivos principales del examen LPI 102:

- 105: Shell, scripts, programación y compilación.
- 106: Entornos gráficos y de escritorio.
- 107: Tareas administrativas.
- 108: Servicios esenciales del sistema.
- 109: Las bases de la red.
- 110: La seguridad.

Este segundo examen versa sobre la administración del sistema y la red de un puesto de trabajo conectado. Son nociones avanzadas: la gestión de los periféricos, la recompilación del núcleo, los módulos, las impresoras, la compartición, la configuración de una red en TCP/IP y los servicios asociados, la seguridad, etc. Así,

determina si usted es capaz de gestionar su sistema y la integración de éste dentro de una red mayor.

Luego se detalla cada objetivo y se pondera según su importancia:

- 104.1: Crear particiones y sistemas de ficheros (2).
- 104.2: La integridad de los sistemas de ficheros (2).
- 104.3: Montar y desmontar sistemas de ficheros (3).
- 104.4: Aplicar cuotas (1).
- 104.5: Modificar los derechos de los ficheros (3).
- 104.6: Crear y modificar los vínculos (2).
- 104.7: Buscar ficheros (2).

El valor entre paréntesis indica la importancia del objetivo y por lo tanto el número de preguntas asociadas en el examen.

## 2. Examinarse

### a. Inscripción

Para poder examinarse de una certificación a través de los exámenes LPI, debe en primer lugar darse de alta en el sitio LPI para obtener una cuenta de usuario. Para ello, diríjase a la dirección siguiente y rellene todos los campos que se le exijan. Trate de no equivocarse en la dirección de su correo electrónico: <https://www.lpi.org/es/register.html>

### b. Matrícula

Cada examen cuesta (en junio de 2011) 145 euros. Por lo tanto, el coste total de la certificación LPIC-1 es de 290 euros si lo obtiene al primer intento. Es posible que durante la celebración de algunos eventos (ferias por ejemplo) encuentre ofertas excepcionales de hasta la mitad del precio.

### c. Centro de examen

Luego debe contactar con un centro de examen para conocer dónde y cuándo se organizan las sesiones. Para ello, puede dirigirse, en España, al sitio LPI-ES: <http://www.lpi.org.es/>.

### d. Desarrollo

El examen se desarrolla en un centro afiliado y, dependiendo de éste, sobre papel o con ayuda de un equipo. Las preguntas suelen estar traducidas al español, pero también es posible que tenga que realizar el examen en inglés. En los dos casos, puede haber tres tipos de preguntas:

- Una única respuesta para dar (escrita o introducida por el usuario).
- Un examen de tipo test con una única respuesta correcta.
- Un examen de tipo test con varias respuestas posibles.

El número de preguntas depende del examen y de la sesión. Cada examen tiene una duración de 90 minutos. Durante este tiempo, está prohibido salir de la sala de examen.

Cuanto más importante sea un objetivo, más preguntas relacionadas con él habrá en la prueba.

#### **e. Aprobado**

Para aprobar un examen debe acertar el 60% de las respuestas. Si efectúa el examen en un equipo, sabrá al finalizarlo si ha aprobado o no. Si lo efectúa sobre papel, tendrá que esperar de tres a cinco semanas antes de recibir el resultado.

# Contenido del libro

Ante la evolución del núcleo Linux a la versión 3.0 publicada en Julio de 2011 y la aparición de nuevas herramientas y de nuevas versiones de distribuciones, nos ha parecido necesario realizar una gran actualización del contenido de este libro, cuyo resultado es esta segunda edición. Aunque el programa de la certificación no ha sufrido modificaciones, sin duda la próxima evolución del examen de certificación tendrá en cuenta estas evoluciones. De este modo, este libro incluye muchos añadidos, algunos pequeños, otros más profundos, para permitir al lector no sólo disponer de todo lo necesario para la certificación, sino de permanecer a la última en lo que a las técnicas de administración Linux en general se refiere.

Este libro abarca todo el programa de la certificación LPIC-1. Sin embargo, en numerosos aspectos va más allá. En ocasiones es recomendable, con el fin de mejorar la comprensión de un aspecto o tema, profundizar o avanzar en los detalles. En otras ocasiones se prefiere regresar a los orígenes, al contexto general, como la historia de Unix, de Linux y de sus distribuciones.

La división de los objetivos de la LPI ha dado lugar a 14 capítulos. Ahora bien, muchos de estos objetivos, sin coincidir entre ellos, forman parte del mismo campo de aplicación. Por ejemplo, los comandos y la programación shell, por guardar cierta relación, han sido agrupados en el mismo capítulo. La lista de objetivos que detallamos a continuación proporciona punto por punto dónde encontrar la información correcta.

El libro sigue una progresión constante y lógica:

- **Capítulo Presentación de Linux:** Unix, Linux, programas libres, compatibilidad de las distribuciones, encontrar ayuda.
- **Capítulo Instalación de Linux y de los paquetes de software:** instalar Linux, programas complementarios mediante paquetes o fuentes y gestionar las bibliotecas.
- **Capítulo El shell y los comandos GNU:** saberlo todo sobre el shell: todos los comandos importantes para trabajar en ficheros, directorios, texto, procesos, etc., para programar scripts y ejecutar sencillas peticiones SQL.
- **Capítulo Los discos y el sistema de ficheros:** saberlo todo sobre los discos, las particiones, los sistemas de ficheros, el swap, las cuotas, los permisos, los puntos de montaje.
- **Capítulo Inicio de Linux, servicios, núcleo y periféricos:** saberlo todo sobre el arranque y parada de Linux: la carga de Linux y de los servicios, la gestión de los módulos, los parámetros del sistema, el núcleo, los periféricos y la gestión del hardware.
- **Capítulo Las tareas administrativas:** la gestión de los usuarios, grupos, contraseñas y de las variables del entorno, las notificaciones, impresiones, automatización de las tareas, la copia de seguridad, la actualización de la hora, la internacionalización.
- **Capítulo La red:** las bases y la configuración TCP/IP de la máquina, los servicios de redes, la configuración básica de Apache, PPP, SSH, DNS, del correo electrónico, de las FTP, de las comparticiones NFS y Samba.
- **Capítulo La seguridad:** los buenos usos en el tema de seguridad, la integridad del sistema, los virus, los rootkits, los límites, las actualizaciones, la seguridad de la red y de los servicios, el firewall, GPG, los derechos SUDO.
- **Capítulo X Window:** instalar y configurar el servidor gráfico X Window, el gestor de visualización y sesiones, elegir y personalizar su entorno, adaptar los programas a sus necesidades (recursos).
- **Capítulo Particionamiento avanzado: RAID y LVM:** configurar una matriz RAID y un LVM, aprender a modificar a voluntad el tamaño de los discos y sistemas de ficheros.

Cada capítulo va acompañado de 30 a 85 preguntas y de 2 a 7 prácticas con todas sus respuestas. En el encabezado, encontrará los requisitos para cada capítulo y la lista de los objetivos.

# Requisitos y objetivos

## 1. Requisitos

- Acceso a una consola y a un navegador para obtener ayuda.

## 2. Objetivos

Al final de este capítulo, usted será capaz:

- De conocer la historia de Unix.
- De conocer la historia de Linux.
- De conocer los grandes principios del programa libre.
- De comprobar la compatibilidad de hardware.
- De elegir una distribución.
- De obtener ayuda en Internet.
- De obtener ayuda desde Linux.

# Bienvenidos al mundo Unix

## 1. Un nuevo mundo

Linux no es un simple producto de la moda ni de una campaña promocional. Desde sus primeros desarrollos en 1991 y hasta la fecha de hoy, Linux no ha dejado de evolucionar, cambiar. El mundo de la informática está vivo. Si no evoluciona, vegeta. Gracias a Linux, millones de personas han encontrado por fin lo que estaban buscando.

Linux no es más complicado que cualquier otro sistema operativo. La lentitud con la que parece penetrar Linux en el gran público no está relacionada con su supuesta dificultad. El buen rendimiento admitido por numerosos usuarios principiantes o avanzados, grupos de usuarios Linux y profesionales en general muestra que se trata sobre todo de un problema relacionado con las costumbres de la gente, habituada durante años a un sistema operativo único. Resulta evidente que hay que modificar un poco estas costumbres para adaptarse a un entorno Linux, de la misma manera que conducir un coche familiar no le convierte en un as de la conducción deportiva con Ferrari.

## 2. Historia de los ordenadores

### a. Complejidad de los ordenadores

Un ordenador es una máquina electrónica extremadamente compleja. Si bien el principio mismo del ordenador tal como lo conocemos no ha cambiado desde la época de Alan Turing o de Conrad Suze y data de principios de los años 40 e incluso antes (máquina de Charles Babbage), las evoluciones tecnológicas y la miniaturización han permitido crear máquinas cada vez más potentes y, al mismo tiempo, pequeñas. Desde los primeros ordenadores electromecánicos compuestos de miles de lámparas de vacío y cuya programación se llevaba a cabo conectando cables al ordenador moderno de hoy, la complejidad del hardware ha sido creciente. Hubo una época en la que un ordenador ocupaba toda una planta y el circuito de agua que permitía enfriarlo (algunos eran enfriados mediante bloques de hielo) calentaba todo un edificio. Hoy en día basta con irse de compras al supermercado del barrio para adquirir un ordenador: se podría pensar de manera lógica que la sencillez en la adquisición del uno de estos equipos traería aparejada la sencillez de uso, y que el ordenador debería ser tan fácil de utilizar como su lector de DVD de salón.

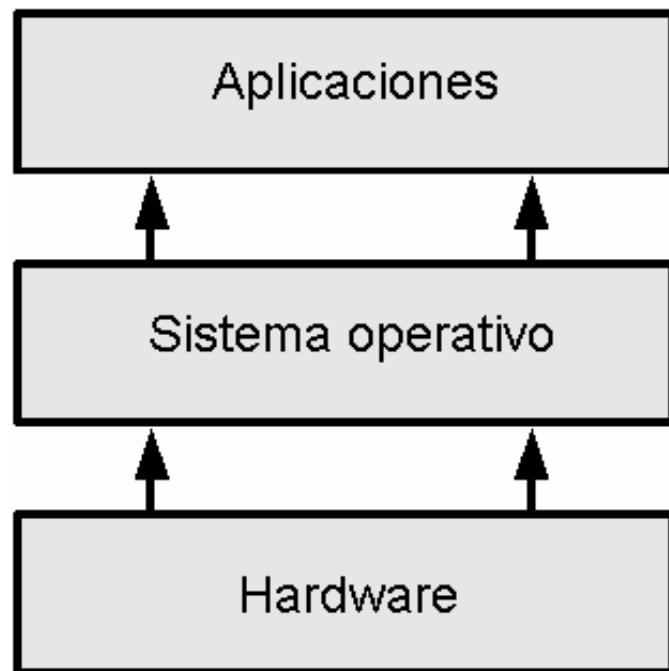
Desafortunadamente, no es el caso. Un ordenador es una máquina muy polivalente: ¿debemos esperar que sea tan sencillo de usar como una máquina de mecanismo simple? ¿Debemos conocer a toda costa los entresijos técnicos de nuestro ordenador para poder utilizarlo? La respuesta suele ser no. Sin embargo, hay que recordar y respetar algunas nociones y bases fundamentales.

### b. La inteligencia

Un ordenador no sabe hacer nada por sí mismo. No hay nada más estúpido que una computadora. Hay que decirle lo que debe hacer. El ordenador es menos inteligente que cualquier insecto. Lo que le hace «inteligente» en el sentido de potencia de cálculo es usted y los programas que le hace ejecutar. Aunque esté inventado por humanos para humanos, el ordenador no entiende su lenguaje. Habla en binario, combinación de ceros (0) y unos (1). Estos valores combinados entre sí forman palabras y datos para el ordenador. El binario se convierte en un lenguaje llamado "lenguaje máquina". Los microprocesadores emplean un lenguaje llamado "ensamblador" en el cual cada instrucción dispone de un equivalente en binario. Con la ayuda de este lenguaje ensamblador, los informáticos desarrollan sus programas, uno de los cuales se llama "compilador": un traductor de lenguaje de alto nivel y comprensible esta vez por un gran número de informáticos.

### 3. El sistema operativo

Entre el momento en el que pulsa el botón de encendido de su ordenador y el momento en el que puede por fin trabajar y utilizar sus programas, pasa un cierto tiempo durante el cual se cargan programas en la memoria de su ordenador. El objetivo de estos programas es simplificarle la vida al usuario y al programador, al hacer las cosas más sencillas. Estos programas forman un conjunto llamado "sistema operativo". Como indica su nombre, el papel del sistema operativo consiste en sacar partido al ordenador tanto como sea posible. En otras palabras, el sistema operativo le proporciona todos los mecanismos necesarios para aprovechar lo mejor posible los recursos de su ordenador.



*Principio del sistema operativo*

---

► Un sistema operativo es un programa o un conjunto de programas que garantiza la gestión del ordenador y de los periféricos. Sirve de interfaz entre el medio físico (hardware) y el programa (software). Es un conjunto de programas muy complejos cuya meta es hacer más sencillos los programas y el uso del ordenador.

---

El sistema operativo ofrece a los programadores una interfaz de programación de aplicaciones llamada **API**, *Application Programming Interface*. Todos los programadores utilizan las mismas funciones en sus programas, lo que simplifica mucho el trabajo. Pueden centrarse en el objetivo de su programa (crear un procesador de texto, por ejemplo) sin tener que escribir una y otra vez las secuencias de programas que gestionan el disco duro, la impresora o el acceso al teclado. Le corresponde al sistema operativo gestionar:

- la memoria,
- los accesos a los periféricos,
- los datos sobre los discos,
- los programas,
- la seguridad,
- la compilación de información.

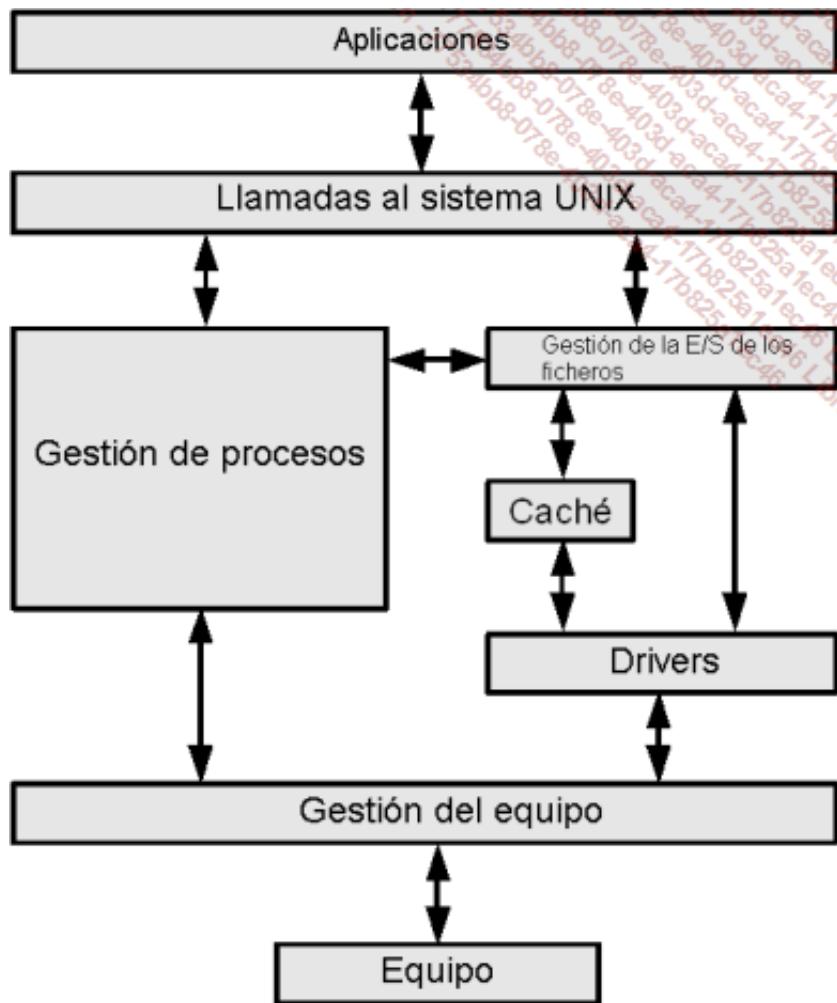
Falta aún la interfaz gráfica. En un producto como Microsoft Windows la interfaz gráfica está incluida en el

el mismo interior del sistema operativo. De hecho, en Windows hay una ventana de diálogo para realizar cualquier ajuste en el sistema. Los usuarios suelen pensar que la interfaz gráfica es el sistema operativo. Sin embargo, la interfaz gráfica no siempre ha formado parte del sistema operativo. Viene como complemento. No encontrará ningún libro sobre teoría de los sistemas operativos que trate de las interfaces gráficas. ¿Por qué? ¿Cuál es el interés, más allá de aumentar la lentitud y ocupar preciosos recursos de la máquina, de tener una interfaz gráfica para gestionar un servidor de Internet? Linux propone interfaces, pero no son programas como los demás.

- La interfaz gráfica no es un componente del sistema operativo y Linux no la necesita para funcionar correctamente. Es un conjunto de varios programas ejecutados de forma clásica "encima" del sistema operativo.

Linux es un sistema operativo de tipo Unix. Existen decenas de sistemas operativos en esta familia. Unix es un sistema operativo de la familia de los sistemas **multitareas** y **multiusuarios**:

- **Multitarea:** el sistema gestiona la ejecución simultánea de varios programas llamados procesos (nota: un verdadero multitarea necesita contar con varios microprocesadores o equivalentes; Hyper Threading, por ejemplo).
- **Multiusuarios:** el sistema permite la existencia de varios usuarios diferentes en una misma máquina, conectados o no (un usuario puede ejecutar un programa sin estar conectado, como por ejemplo un servidor de Internet).



Arquitectura lógica de un sistema UNIX

El esquema anterior es una síntesis simplificada de la estructura interna de un sistema operativo Unix. Abajo

se encuentra su medio físico; arriba, los programas que usted ejecuta en su máquina. Entre ambos, los diferentes componentes del sistema operativo aseguran su correcto funcionamiento:

- Las **llamadas al sistema** son utilizadas por los programas para comunicarse con el sistema operativo Unix.
- La **gestión de los procesos** se encarga de la conmutación de las tareas y de su prioridad. Este componente se encarga, por lo tanto, de la multitarea.
- La **gestión de las entradas y de las salidas de ficheros** se encarga tanto de la lectura y escritura de los datos en sus discos duros como también en sus periféricos (tarjeta de sonido, impresora, etc.).
- Se puede ubicar cierta información en una zona de memoria llamada **caché**. En lugar de escribir datos directamente en el disco duro (un proceso lento), Unix va a escribirlos provisionalmente en una zona de memoria para luego escribirlos en el disco, después de unos segundos. Así, la lectura de estos datos es más rápida porque ya están en memoria y el programa no pierde tiempo en esperar el fin de la escritura de los datos.
- Los **drivers** tienen como papel gestionar al más bajo nivel el hardware o las estructuras lógicas del hardware (por ejemplo, los datos de una partición).

---

 Una aplicación bien programada en un sistema operativo bien programado no puede cortocircuitar este esquema: no "discute" nunca con el hardware, sino que dialoga obligatoriamente con las API correspondientes.

---

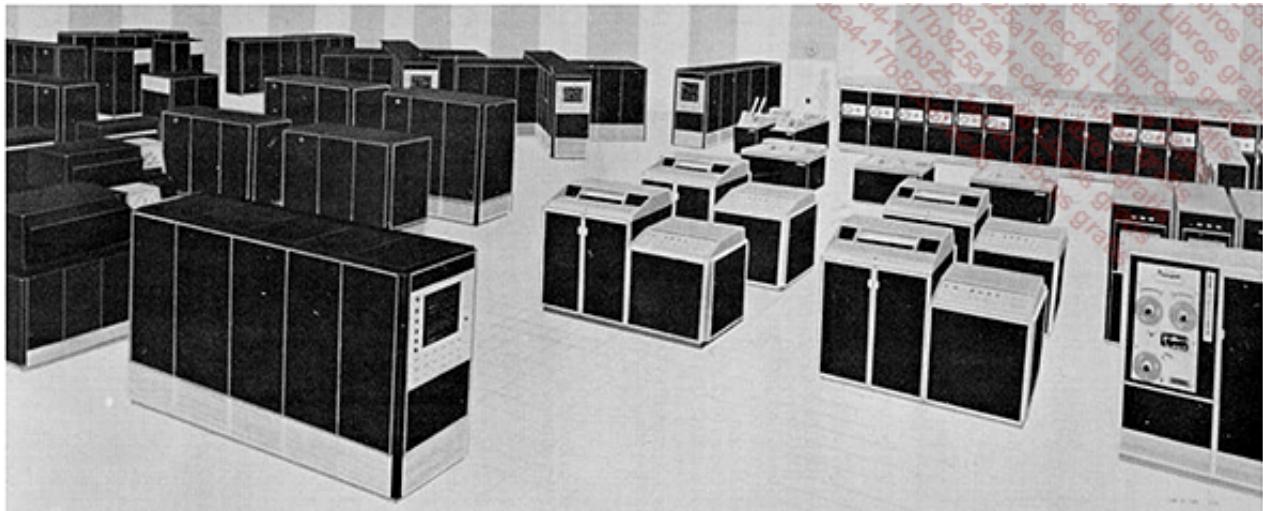
## 4. El sistema Unix, una breve historia

### a. De MULTICS a UNIX

La historia de Unix empieza en **1964**, cuando el MIT, el laboratorio Bell Labs de AT&T y General Electric comienzan a desarrollar el proyecto experimental **MULTICS** (*Multiplexed Information and Computing Service*). El proyecto Multics responde a estas nuevas necesidades:

- posibilidad de ser utilizado por varias personas a la vez,
- posibilidad de lanzar procesos en segundo plano,
- una gestión más adecuada de la seguridad.

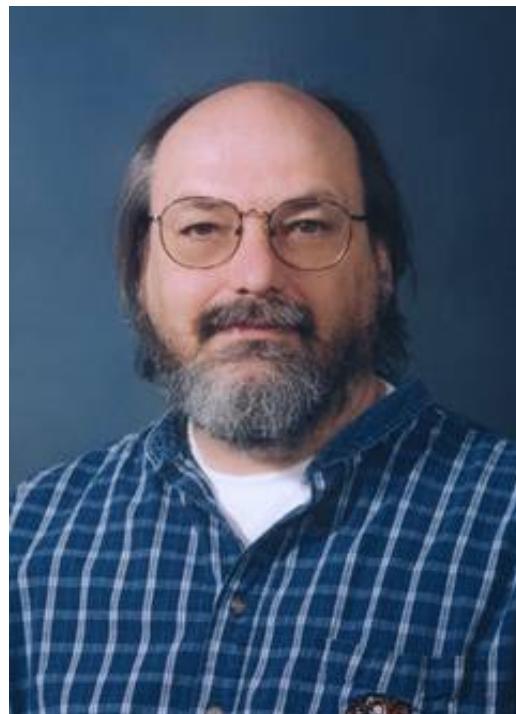
Multics se desarrolla en un enorme sistema GE-645 de General Electric, equipado con dos procesadores que pueden tratar cada uno 435 000 instrucciones por segundo, tres unidades de memoria de 1 MB cada una y 136 MB de almacenamiento. Funcionó en el MIT hasta 1988, con 82 puestos de trabajo, y un máximo de 200 usuarios llegó a trabajar de forma simultánea en General Electric. La última instalación Multics que ha sido desactivada es la del Ministerio de Defensa Canadiense, el 30 de octubre de 2000.



*El GE-645*

Sin embargo, si bien Multics alcanzó pronto un grado de estabilidad lo suficientemente aceptable como para pasar a producción, resultó que su rendimiento era menor de lo esperado. En **1969** Bell Labs abandona el proyecto para dedicarse al desarrollo de otro sistema llamado GECOS.

**Ken Thompson**, desarrollador en Bell, continuó trabajando en el GE-645 y se propuso la creación de un juego llamado Space Travel. Al ejecutarse en Multics resultó ser muy lento y costoso, más si se le medía en tiempo compartido. En este modelo, el tiempo de procesador era subdividido en secciones y cada una se destinaba a cada proceso en memoria.



*Ken Thompson*

Ken reescribe entonces el juego en un ensamblador para el miniordenador DEC PDP-7. En esta misión le ayuda **Dennis Ritchie**, procedente también de Bell Labs. Esta experiencia, combinada con la obtenida durante la concepción de Multics, empuja a los dos hombres y a su equipo a crear un nuevo sistema operativo para el PDP-7. **Rudd Canaday**, también de Bell Labs, estaba desarrollando justamente un nuevo sistema de ficheros que concibe como si fuera un sistema operativo; de ahí que Unix sea un sistema

orientado a ficheros, en el cual todo (o casi) es fichero. Le añadieron un intérprete de comandos y algunas herramientas. Llamaron al sistema **UNICS** (*Uniplexed Information and Computing System*), según una idea de **Brian Kernighan**. Desde el principio el proyecto podía ya gestionar dos usuarios al mismo tiempo en modo multitarea real.



El DEC PDP-7

---

► El origen de la palabra es tema de numerosas leyendas, todas las cuales tienen probablemente su parte de verdad. UNICS es un juego de palabras en clave de humor de MULTICS, cuya arquitectura es el motivo de numerosas críticas en esta época: «MULTICS (múltiple) hacía la misma cosa de varias maneras, mientras UNICS (único) hacía cada cosa de la misma manera». Además, en inglés UNICS se pronuncia como «eunuchs», o sea, «eunuco», un sistema Multics «castrado».

---

UNICS retoma los conceptos esenciales desarrollados para MULTICS y los mejora. En particular, sus diseñadores proponen un sistema de comunicación completamente nuevo entre los programas, en el cual un primer programa puede enviar sus datos a otro programa. Rápidamente, el CS fue sustituido por un X, una letra menos para la misma presentación. La leyenda **UNIX** acababa de nacer.

Todo hubiera podido finalizar aquí, ya que los equipos trabajaban sin financiación. Bell Labs se deshizo de Multics y de sus sucesores. Para proseguir los trabajos, Thompson y Ritchie proponen a Bell Labs añadir un programa de procesador de textos para el PDP-11/20 en UNIX. La compañía acepta, se pone a disposición la máquina y el equipo obtiene una financiación y un soporte oficiales. La herramienta *runoff* (que se convertirá en roff y luego en troff) y el editor ed son desarrollados y por primera vez en **1970** se emplea la denominación **Unix Operating System**. Bell utiliza entonces Unix como un sistema de procesador de texto para la redacción de sus patentes. El primer manual de programación Unix data del 3 de noviembre de 1971.

## b. El lenguaje C

Rápidamente, los ingenieros deben enfrentarse a un nuevo problema. Desarrollado en un ensamblador y por lo tanto en lenguaje máquina, Unix debe ser reescrito en parte para cada nuevo modelo de ordenador DEC, y programar en ensamblador es un arte difícil. Es el momento de enfrentarse al reto de la portabilidad. En 1970, Thompson se interesa por el problema. Primero piensa en desarrollar Unix en lenguaje TMG y luego en Fortran. Al encontrar el lenguaje incompleto, se asocia con Dennis Ritchie para crear el **lenguaje B**, que proviene del lenguaje **BCPL**. Pero tampoco les parece el más adecuado (problema con el tipado de las variables y los números reales). Ritchie parte del lenguaje B y desarrolla el **New B**, que llama lógicamente el **lenguaje C**. C es volcado a lenguaje máquina tras pasar una etapa de compilación. La escritura de los programas resulta más rápida.



Dennis Ritchie

Unix es reescrito en lenguaje C a partir de **1973**. Para pasar un sistema Unix de una máquina a otra, basta con que haya un compilador C instalado en la nueva máquina. Es mucho más sencillo y eficiente escribir un compilador C (él mismo escrito en gran parte en C) que escribir de nuevo todo un sistema operativo en ensamblador. Únicamente los elementos que tenían mayor contacto con la arquitectura física de la máquina fueron escritos directamente en lenguaje máquina. Unix se vuelve portable y eso permite acelerar su desarrollo.

## c. Las licencias y el advenimiento de BSD y System V

Un hecho de gran calado va a contribuir en aquel entonces a consolidar la gran difusión de Unix (la palabra "gran" puede parecer exagerada cuando se está hablando de decenas de copias del sistema). AT&T, de la cual depende Bell Labs, fue objeto en 1956 de un decreto antimonopolio que le prohibía comercializar cualquier producto ajeno a su sector industrial: las telecomunicaciones. No puede vender Unix. AT&T (que ni siquiera le ve futuro comercial) decide en **1974** liberar completamente el sistema UNIX, pero sólo con fines educativos a las universidades y las empresas y bajo una licencia que resulta ser muy poco restrictiva. Sólo el código fuente (*el texto del programa no compilado aún*) del núcleo en ensamblador no será incluido en la liberación, al menos oficialmente. UNIX cuenta ya con algunas versiones, de entre cuales las más difundidas son la sexta, de **1975**, y la séptima, de **1978**. La siguiente versión, Unix v7, fue la primera específicamente diseñada con la intención manifiesta de ser llevada a otras máquinas además de las PDP, en particular al modelo VAX 11/780. Se considera que la v7 fue la última versión completamente común a todos los Unix siguientes.

Por esta época se produce otro gran acontecimiento. Unix iba a celebrar sus diez años y las universidades

estadounidenses se estaban implicando con fuerza en su difusión y mejora cuando AT&T modifica la licencia de Unix haciéndola más restrictiva. La entidad que comercializa Unix tiene la autorización de vender licencias del código fuente. Las tarifas, prohibitivas, obligan a las universidades a continuar, para lo bueno y lo malo, sus desarrollos tomando como punto de partida las versiones anteriores al cambio de licencia. Una de estas universidades se encuentra en California y se llama **Berkeley**. Berkeley es la mayor contribuidora de Unix desde que empezó a trabajar en él, en 1974. La primera versión UNIX de **BSD** (*Berkeley Software Distribution*) está basada en Unix v6, de **1977**, y recibe el nombre comercial de **1BSD**. La versión **2BSD** está basada en Unix v7 y data de 1978.

A partir de ahí, se asiste al enfrentamiento de dos escuelas UNIX. La primera, en teoría la oficial, es la de AT&T, que va a seguir desarrollando las versiones 8, 9 y 10 durante los años 80 del siglo XX, con fines científicos. De forma paralela, desarrolla un Unix totalmente comercial llamado **Unix System III**, que se comercializa a partir de **1982**. En **1983** AT&T desarrolla y vende las primeras versiones Unix System V. La última versión, **Unix System V release 4.2**, data de **1993**. La historiografía de Unix conoce esta versión con la abreviatura SVR4 y su principal característica es que su código fuente está disponible bajo licencia. Esto significa que un organismo está autorizado a comprar una y a desarrollar su propia versión comercial.

Durante este tiempo, la universidad de Berkeley no descansa y sigue desarrollando BSD como alternativa Open Source al Unix System III y V, cuyas fuentes no puede usar por carecer de derechos. Es en BSD donde se va a implementar por primera vez el protocolo **TCP/IP**, base del Internet moderno, mediante la financiación del Ministerio estadounidense de Defensa. La última versión oficial de BSD es la **4.4BSD** y data de junio de **1994**.

#### d. La guerra de los Unix

El período que va desde la mitad de los años 80 hasta 1994 no ha estado exento de sobresaltos. Los efectos de la separación de Unix en dos ramas ha resultado ser un desastre y a punto ha estado de significar la ruina. Los dos campos (AT&T con su System V y Berkeley con su BSD) no se ponen de acuerdo en un estándar común. La consecuencia, además de la celebración de múltiples juicios (hasta 1993) sobre el uso del nombre y herramientas derivados de Unix, es el hecho de que han crecido como la mala hierba multitud de versiones de Unix comerciales y sobre todo incompatible entre ellas. Es a partir de esta época cuando surgen los grandes nombres de los clones Unix, entre los cuales cabe destacar **Solaris**, **AIX**, **OSF1 / Digital Unix / True64**, **Xenix**, **HP-UX**, **IRIX**, **Ultrix**, **Unixware**, **A/UX**. Todos son incompatibles con el vecino, pero claman alto y claro su pertenencia a Unix. Esta guerra de los Unix se conoce realmente como el período oscuro de las *Unix wars*. Pero lo más triste es que nadie parece interesado en ponerse de acuerdo sobre una base y un estándar común. El efecto directo de esta guerra ha sido la creación de un vacío en el mercado aprovechado por Microsoft para colocar con fuerza su sistema operativo **Windows NT** (que a su vez, pero poca gente lo sabe, deriva de Unix).

En **1984** un grupo de editores comerciales de Unix intenta una primera estandarización creando X/Open Standards y publicando un documento llamado **X/Open Portability Guide**, que describe un estándar abierto (accesible a todos) para Unix. Este comité obtiene un gran espaldarazo en **1987**, cuando Sun Microsystems y AT&T deciden trabajar sobre un Unix unificado, fusión de BSD y de System V. El resultado es en realidad System V Release 4.

El refrán dice que pican más los celos que las pulgas. La competencia acusa a Sun de querer convertirse en el maestro del juego y funda *Open Software Foundation*, o sea, **OSF**, en **1988**. OSF declara que también persigue un estándar abierto para Unix, pero basado de manera exclusiva en BSD. Publican sus especificaciones en **1990**.

Como respuesta, AT&T y su grupo crean *Unix International* en **1989**, en un enésimo intento de unificación, sin éxito. En medio de este embrollo, AT&T decide deshacerse de Unix, del que sigue siendo propietaria oficialmente, y crea para ello una empresa llamada *Unix System Laboratories* en **1992**. Todos los derechos de Unix se transfieren a USL.

#### e. La estandarización

Un año más tarde, y cuando la situación amenaza con estancarse, aparece un nuevo actor que pretende triunfar donde los otros fracasaron. La empresa **Novell** compra USL el año de su creación y se convierte en el propietario de SVR4.2. En **1993** Novell ceda la marca Unix a X/Open. Unix International desaparece en **1994** y se reestructura OSF. Finalmente, en **1995** Novell cede la licencia operativa del código fuente de Unix a la empresa **SCOSanta Cruz Operations** (que se convertirá en Caldera y luego, de nuevo, en SCO). El mismo año X/Open y OSF se fusionan de manera definitiva y forman **The Open Group**.

Ya sólo existe un único organismo de estándar Unix. Unix es, por lo tanto, un sistema operativo abierto: sus especificaciones son conocidas y cada editor de sistema Unix comercial o gratuito que desea asegurar una compatibilidad con el conjunto de los Unix debe implementar este estándar, aunque sigue siendo libre de programar este estándar como deseé, ya que una misma función puede ser escrita de varias maneras. Los "estragos" de las *Unix Wars* han sido numerosos y han permitido el surgimiento de otros sistemas operativos como Windows NT de Microsoft.

## f. Unix es un estándar

Para asegurarse de que todos los Unix siguen las mismas recomendaciones, The Open Group publica las normas (*Single Unix Specification, Unix95, Unix98, Linux Standard Base, etc.*) y puede proponer certificaciones. Estas normas se apoyan, en parte, sobre las definidas desde **1988** por el **IEEE Institute of Electrical and Electronics Engineers** (que se pronuncia "I3E") y en particular en la **IEEE 1003**, llamada también **POSIX** (*Portable Operating System Interface*). La X es al mismo tiempo una herencia y un reconocimiento al trabajo efectuado en Unix. IEEE 1003 se compone de 15 documentos que agrupan por temas todo lo que debe contener un Unix (comandos básicos, interpretador de comandos, interfaces de usuario, funciones de programación, etc.) para estar conforme al estándar POSIX. POSIX no se limita a Unix. Windows NT es conforme a POSIX para algunos de sus componentes. POSIX no es un estándar abierto. Las especificaciones de *The Open Group* están abiertas y accesibles a todos y los editores prefieren referirse a ellas.

Para poder utilizarse en algunas administraciones estadounidenses, un sistema operativo Unix debe ser conforme al estándar POSIX. De hecho, cuando Linux tuvo que usarse, el gobierno de Bill Clinton hizo financiar la totalidad de la certificación **PCTS** (*Posix Conformance Test Suite*) por el Tesoro estadounidense.



Las últimas versiones oficiales de BSD y System V datan de 1994. Los Unix concebidos a partir de 1995 implementan las recomendaciones de The Open Group. Sin embargo, históricamente algunas versiones siguen más «orientadas» a BSD, o más a System V en lo que se refiere a su configuración, o a veces a los dos, como Linux (y según la distribución).

## g. Unix en los ordenadores personales

El primer Unix para ordenador personal, en el sentido de ordenador de tipo IBM PC es **Xenix**. Proviene de Unix v7 y salió en **1983** para PC (algunas versiones han estado disponibles incluso antes en otras arquitecturas físicas). Microsoft ha llevado Xenix, a costa de numerosas modificaciones, a otras máquinas. La versión 2 de Xenix data de 1985 y está basada en Unix System V. Cuando en 1987 IBM lanzó el sistema operativo OS/2 en asociación con Microsoft, este último transfirió los derechos de Xenix a SCO. La versión 2.3.1 de este mismo año soporta el 386, SCSI y TCP/IP. Xenix se convierte en SCO Unix en 1989 y luego desaparece en favor de SVR4.



*Andrew Tanenbaum*

El norteamericano **Andrew Stuart Tanenbaum** (apodado Andy) es investigador y profesor de informática de la Universidad libre de Ámsterdam. También es autor de obras de referencia en informática sobre la teoría de los sistemas operativos. En **1987**, con fines pedagógicos, concibe y escribe el sistema operativo Minix. Utiliza 20 MB de espacio en disco y necesita pocos recursos (2 MB de memoria física). Minix tendrá una gran importancia para Linux. Minix sigue existiendo y la versión 3 salió en el octubre de 2005.

Hay otras versiones de BSD adaptadas al PC. Una de ellas es el **386BSD**, de octubre de 1989 y que deriva de 4BSD. Su sucesor más famoso es el **NetBSD**, que deriva de 4.3BSD y de 4.4BSD. NetBSD es el sistema operativo más portado a otras arquitecturas físicas. **OpenBSD** está basado en 4.4BSD y muy orientado a la seguridad: en ocho años sólo se pudo encontrar un fallo de seguridad. **FreeBSD** deriva también de 4.4BSD y proviene directamente de la época de los juicios entre BSDI y AT&T. Gracias a ello, FreeBSD es completamente libre y abierto.

**Solaris**, el Unix de Sun Microsystems, está disponible desde hace varios años en PC y la versión OpenSolaris (versión 10) es Open Source.

En fin, **Linux** es sin duda el Unix libre más conocido y más extendido en el PC. La historia de su creación merece detenerse un poco en sus detalles.

# El software libre

## 1. Los orígenes del software libre

Unix es el perfecto ejemplo del trabajo que se puede efectuar cuando se encauzan todas las energías a la búsqueda de un ideal tecnológico. Cuando AT&T distribuye casi libremente en 1974 el código fuente del sistema operativo a las universidades porque, entre otras razones, no ve ningún futuro económico a su producto, no parece dudar del entusiasmo de los estudiantes, profesores e investigadores en informática. Esta primera comunidad pasará mucho tiempo modificando y mejorando el producto, subiendo todas las novedades a AT&T para que se integren al producto oficial. Tras el cambio de licencia en 1978, la energía de la comunidad se encauzó hacia el proyecto universitario BSD, dejando el Unix comercial de AT&T. Señalemos que los más grandes progresos se hicieron con el Unix de Berkeley.

Los primeros ordenadores eran esencialmente herramientas de búsqueda en manos de universitarios (y también monstruos de cálculo para necesidades militares). En los laboratorios de investigación, los programas circulaban como las ideas: libremente. Era absolutamente normal que un programa desarrollado por un equipo de programadores o investigadores se distribuyera a otros equipos de otras universidades y a cualquier otro lugar donde hiciera falta. Y nada había de raro en que este programa fuera modificado por otro equipo, y así sucesivamente. A día de hoy, cuando un ilustre matemático demuestra un teorema difícil, publica el resultado de sus investigaciones en obras especializadas con el fin de ayudar al progreso de la ciencia. Todo el mundo tiene acceso a ello.

Pero el universo de la informática ha seguido otros derroteros. Pese a ser una ciencia, el fruto de las investigaciones en informática no se circumscribe al mundo de los universitarios. Rápidamente, las empresas vieron el inmenso interés de automatizar algunas de sus tareas, como la contabilidad, los pagos, etc. Con la compra de los primeros grandes ordenadores de gestión, se necesitaron programas. Estos programas tuvieron que ser protegidos como secretos industriales: había nacido una nueva industria: la creación de programas. Con su entrada en la dinámica de las grandes empresas, la informática perdió rápidamente la inocencia y se hizo mucho menos libre. Se empezó a hablar de licencias, impuestos y tasas, derechos de autor (lo que no impide autorizar la copia según el caso), limitación de los derechos, prohibición de copiar, etc.

## 2. El proyecto GNU y la FSF

**Richard Stallman** no fue seguramente el primero en lamentar este hecho, pero decidió reaccionar. Informático en el laboratorio de inteligencia artificial en el MIT a finales de los años 1970, emplea una impresora que se avería a menudo. Como él y sus compañeros disponían del código fuente del driver (programa de gestión) de la impresora, lo modificaron para recibir una señal de cada avería. En un momento dado, el laboratorio compra un nuevo modelo de Xerox más fiable, pero el driver para su sistema operativo no aparece. Como desea adaptar este driver a sus necesidades, Richard Stallman, recurre a otro laboratorio que dispone del código fuente, pero que se niega a proporcionárselo: Xerox lo prohíbe. Esto significa que la impresora nunca funcionará, y Stallman, muy contrariado por esta situación, decide obrar en pro de la defensa y la difusión del software libre enfrentándose al cerrado mundo del software propietario.



*Richard Stallman*

Stallman decide en **1983** escribir un nuevo sistema operativo de acceso, uso, modificación y redistribución completamente libres. Basado en Unix, lo nombra **GNU** (*Gnu's Not Unix*). Los acrónimos recurrentes estaban muy de moda entre los informáticos. Puede encontrarse el anuncio del proyecto y de las motivaciones de Stallman en <http://www.gnu.org/gnu/initial-announcement.html>. Para su sistema necesita un núcleo (el corazón del sistema operativo) y herramientas (para gestionar los ficheros, por ejemplo). Pero no se trata de su primera intentona, ya que escribió un gran editor de texto llamado **Emacs**. Los primeros desarrollos van muy deprisa y las herramientas son muy numerosas y a menudo de mejor calidad que las comerciales. A cambio, el diseño de un núcleo Unix es mucho más complejo y necesita una fase teórica importante. Se lanza el proyecto **HURD** (*Hurd of Unix Replacing Daemons*), que aún no ha finalizado.

La batalla no es solamente técnica, sino también política, filosófica, comercial y jurídica. Para defender el software libre, Stallman crea la **FSF** (*Free Software Foundation*) en 1985, que publica las ideas del software libre. Entre sus primeros trabajos figura la redacción (con la ayuda de abogados) de una licencia especial para estos programas llamada la **GPL** (*General Public License*). Un software libre garantiza cuatro libertades:

- **Libertad 0:** la libertad de utilizar un programa, sea cual sea el uso que se haga de él.
- **Libertad 1:** la libertad de estudiar el funcionamiento del programa y adaptarlo a sus necesidades.
- **Libertad 2:** la libertad de redistribuir copias con el fin de ayudar a su vecino (en el sentido amplio del término).
- **Libertad 3:** la libertad de mejorar el programa y difundir las mejoras al público general con el fin de que pueda aprovecharlo la comunidad.

Las libertades 1 y 3 necesitan tener obligatoriamente acceso al código fuente del programa. La libertad 3 define la noción de comunidad alrededor del software libre.

---

► Observe que la palabra «gratuito» no se indica en ninguna parte. En inglés «free» significa tanto libre como gratuito. Se debe entender el software libre en el sentido de «libertad», y no de «gratuidad» (*Free as a speech* y no *Free as a beer*, como dicen los ingleses). Es totalmente posible, y a veces aconsejable, empezar con el software libre. Pero como las libertades 2 y 3 autorizan la publicación del programa, siempre es posible recuperar una copia de ello gratuitamente, y eso de manera legal. La gratuidad es un efecto de la libertad tal como está definida para el software libre.

---

Los trabajos de HURD avanzan poco o mal. Sus desarrolladores han apostado por desarrollar un micronúcleo, mientras que los subsistemas se hallan «esparcidos» en varias unidades independientes, pero que deben comunicarse entre ellas. La elección teórica es excelente, pero la implementación técnica es muy difícil. GNU no dispone de núcleo. Es Linux quien va a finalizar el proyecto en **1992**, cuando su licencia se convierte al GPL.

### 3. El Open Source

¿Y qué pasa con el **Open Source**? La expresión apareció en 1998 cuando Netscape Communicator se convirtió en un programa libre. La expresión *Open Source* (fuente abierta) era utilizada en las campañas publicitarias que intentaban llegar a las empresas con la publicación del código fuente libre. El reto consistía en hacer abstracción de las aportaciones fundamentales del código fuente libre para centrarse únicamente en las ventajas técnicas y económicas de este nuevo modelo. Con el tiempo, la expresión ha sido retomada en todos los sentidos por los medios de divulgación y las empresas, y su definición se ha «emborronado». Se habló de «Open Source limitado», proponiendo el acceso a las fuentes pero sin derecho de modificación o de redistribución. En nuestra opinión, el programa libre no debe mostrar medias tintas. O es libre o no lo es.

---

-  Si usted quiere estar seguro de que el programa que utiliza es libre, verifique el nombre de la licencia y diríjase al sitio de **OSI** «*Open Source Initiative*», <http://www.opensource.org>, donde se recogen la mayoría de los más conocidos. Es una iniciativa de **Eric S. Raymond** (ESR), hacker (especialista de alto nivel) y uno de los grandes nombres del Open Source. A veces en conflicto con Richard Stallman, sus dos visiones (técnicas para ESR, filosóficas para Stallman) son, sin embargo, complementarias.
- 

### 4. GNU/Linux

#### a. Linus Torvalds

La historia de Linux empieza cuando Linus Torvalds, joven estudiante finés en la universidad de Helsinki de 21 años de edad, adquiere en 1991 un ordenador 386 para sustituir su Sinclair QL que empieza a mostrar limitaciones. El 386 es un microprocesador de 32 bits genial que gestiona, entre otras cosas, la memoria virtual y la conmutación de las tareas. Pero el gran problema es que recibe un PC con MS-DOS, un sistema operativo que está lejos de ser ideal y sobre todo que no saca el rendimiento adecuado a este procesador. Linus tuvo entonces la idea de instalar otro sistema llamado Minix, un pequeño Unix sencillo y gratis desarrollado por el famoso Andrew Tanenbaum, que sí aprovechaba su bonito PC recién comprado con un crédito. Linus se pone a trabajar y a desarrollarlo. Su objetivo es aprender el funcionamiento del 386, en particular la conmutación de las tareas en lenguaje ensamblador. Empieza a trabajar en un proyecto bastante sencillo: un emulador de terminal, completamente en ensamblador, para conectarse al servidor de su universidad.



*Linus Torvalds*

### **b. El accidente**

Pero he aquí que un día borra accidentalmente los primeros sectores de la partición de su disco duro que contiene Minix, borrando por lo tanto su principal herramienta de desarrollo. Sólo le quedan dos soluciones: o reinstalarlo todo, o partir de lo aún queda y tratar de mejorarlo y aumentar su autonomía. Por supuesto, el entorno de desarrollo es reinstalado, pero Linus decide mejorar su proyecto y añadirle lo necesario: código básico, driver rudimentario de disco duro, paso al lenguaje C, etc. El 25 de agosto de 1991, la versión 0.01 está lista y se publica ante una casi total indiferencia. Para las herramientas, nada más sencillo: el proyecto GNU iniciado por Richard Stallman dispone ya de todo lo necesario. Linux será el núcleo que falta al sistema operativo GNU.

### **c. La primera versión oficial**

El objetivo de Linux es superar a Minix. Ya en su primera versión, es preciso que cuente con un shell (intérprete de comandos) y el gcc (compilador C). Esto lo consigue con la versión 0.02, anunciada en estos términos el 5 de octubre de 1991 en el grupo com.os.minix:

*«¿Echa de menos los días felices de Minix-1.1, época bendita, cuando los hombres eran dignos de este nombre y escribían sus propios drivers de periféricos? ¿Quiere comprometerse en un proyecto original y está deseando encontrar un sistema modifiable a su conveniencia? ¿Está deseando que todo funcione bajo Minix? ¿Echa de menos las noches en blanco intentando implementar un programa recalcitrante? Si es así, lea lo que sigue: como ya he publicado hace un mes, estoy trabajando actualmente en una versión libre de un sistema similar a Minix para un ordenador AT-386. Este sistema ya está operativo (aunque tal vez no le*

*convenga, todo depende de lo que esté buscando), y pienso publicar sus fuentes. Se trata, de momento, de la versión 0.02, capaz sin embargo de ejecutar bash, gcc, gnu-make, gnu-sed, compress, etc.»*

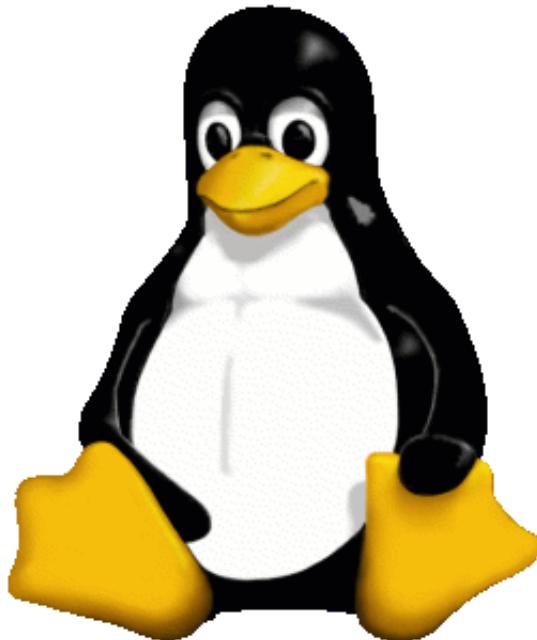
#### **d. El éxito comunitario**

A partir de este momento y gracias a la publicación por Internet, Linux conoce el éxito y las contribuciones empiezan a llegar. Nace la comunidad Linux. La versión 0.03 ve la luz, y luego la 0.10. En 1992, Linux puede por fin hacer funcionar la interfaz gráfica X11. De hecho, se cree que es momento de pisar el acelerador y la versión siguiente recibe el nombre de 0.99 pensando que ya ha llegado a la meta. Fue un error. En efecto, hubo que esperar 18 meses antes de que la versión 0.99pl114 estuviese finalizada y de que por fin la versión 1.0 saliera, en enero de 1994. Entre los primeros balbuceos y la versión 1.0 hubo algunas sorpresas, en particular un conflicto «técnico» entre la concepción monolítica de Linux y la visión micro-núcleo de Andrew Tanenbaum. Este último comenzó las hostilidades con la famosa frase: «Linux está obsoleto».

#### **e. Los años 1994-1997**

Estos años ven aparecer las grandes distribuciones Linux que conocemos todavía hoy en día: Red Hat, Debian, Suse, Slackware. Mandriva (anteriormente Mandrake) llegó un poco más tarde. Durante estos años, Linux no deja de mejorar, con la llegada de la modularidad y de la versión 2.0. Es sobre todo durante estos años cuando Linux sale del pequeño mundo de los hackers y se da a conocer en las empresas. Los proyectos abundan, y la idea de mejorar el sistema y prepararlo para el mundo del escritorio (desktop) se abre camino con el principio del desarrollo de productos como Gnome o KDE.

La mascota de Linux, llamada **Tux**, data de 1996 y fue creada por Larry Ewing con la ayuda del programa libre GIMP. Tux (apócope de *Tuxedo* y *Torvalds UniX*) no es un pingüino, sino un pájaro bobo pigmeo. El hecho es que la palabra inglesa «penguin» designa en este idioma tanto al verdadero pingüino (*razorbill*) como al pájaro bobo; de ahí la confusión.



*Tux, la mascota de Linux*

#### **f. A partir de 1998: la explosión**

No se sabe si se debe a un cansancio general de los usuarios, pero 1998 es el año de los anuncios espectaculares. El mundo de la informática se da cuenta por fin de que Linux no es un juguete para estudiantes manitas. En enero de 1998, Netscape anuncia que su producto pasa a Open Source. De ahí saldrán Mozilla, Firefox y Thunderbird. Los institutos de formación añaden Linux a su catálogo. En julio de 1998, Oracle e Informix son adaptados a Linux. En septiembre, IBM lleva DB2, y Sybase hace lo mismo. Linus Torvalds aparece en la portada de *Forbes*. KDE y Gnome llegan a la versión 1.0. En la bolsa, las cotizaciones suben, las empresas Linux nacen. Es el éxito.

En enero de 1999, llega Linux 2.2 y su imparable éxito hace reaccionar a Microsoft. Es David contra Goliat y sigue siendo así. Se hubiera podido pensar que la explosión de la burbuja de Internet en la bolsa en 2000 lo habría hecho fracasar todo. Pero ya se sabe que no. Linux no es un coloso de pies de barro. Sus pies son la comunidad, inquebrantable. El núcleo 2.4 sale el 4 de enero de 2001. El núcleo 2.6 sale el 18 de diciembre de 2003; el 3.0 en julio de 2011.

### **g. Hoy y mañana**

Hoy Linux es famoso por ser un sistema operativo estable, fuerte y competente. Se utiliza en más de un tercio de los servidores en el mundo y en la mitad de los servidores Web. Conquistó el mundo de la empresa, el mundo universitario. Sobre todo, supo conservar su independencia, garantizada por la comunidad y el número de contribuyentes, frente a los gigantes de la informática. El próximo reto importante para Linux es el entorno de escritorio y, por qué no, el hogar, en sustitución del Windows. Queda todavía camino por recorrer, pero muchos son los que ya han dado el primer paso.

Es en otros ámbitos donde Linux se impone y tiene un uso habitual. Incluso se puede afirmar que decenas de millones de personas lo utilizan cotidianamente sin saberlo. Los distintos dispositivos de los proveedores de acceso a Internet funcionan con Linux. También la gran mayoría de los dispositivos multimedia. Cualquier usuario que consulte las propiedades de su teléfono móvil o de su tableta Android podrá comprobar que es un núcleo Linux el que las hace funcionar. Lo mismo sucede con los GPS de marca Tomtom, entre otros.

# ¿Qué hardware es compatible con Linux?

## 1. La arquitectura

Linux es compatible con al menos tres arquitecturas físicas corrientes:

- **x86** para los ordenadores cuyos procesadores son del tipo Intel (del 386 al Pentium 4) o AMD (Athlon, Duron, Sempron) de 32 bits. Esta versión funciona también en las máquinas con procesadores de 64 bits.
- **x86\_64** para los ordenadores cuyos procesadores son del tipo Intel (Pentium 4 a partir de las series 600, Xeon, Dual Core/Quad Core) o AMD (Athlon 64, Sempron 64, Opteron) de 64 bits. Esta versión no funciona en los procesadores de 32 bits.
- **ppc** para los ordenadores cuyos procesadores son de tipo PowerPC, o sea, los antiguos ordenadores de la marca Apple. Esta versión no se instalará en las últimas máquinas Apple, basadas en un procesador Intel. Aún existen algunas distribuciones para esta arquitectura.
- **arm**: es un caso más particular, ya que esta familia de procesadores se usa esencialmente en sistemas embedidos, especialmente en dispositivos multimedia, en "cajas" multifunción, en routers, en lectores DVD, DivX y Blu-ray de salón, en GPS o en smartphones y tabletas.

 Hace algún tiempo algunos drivers o aplicaciones no estaban adaptados a los procesadores o distribuciones de 64 bits. Hoy en día esto ya no sucede, ya que casi todos los procesadores y los sistemas operativos son de 64 bits, incluyendo Windows (desde hace ya tiempo). Sin embargo, un sistema operativo de 32 bits funcionará perfectamente en una arquitectura Intel 64 bits. Si usted detecta disfunciones molestas y problemas en periféricos, considere instalar una versión de 32 bits que podría resolver sus problemas.

### Configuración física básica

Linux soporta en teoría todos los tipos de procesadores desde la versión 386, y puede funcionar con sólo unos megabytes de memoria. La distribución Polux Linux funciona en un 386 con 4 MB de memoria. La distribución Damn Small Linux funciona con un 486, 16 MB de memoria y utiliza 50 MB de espacio en disco. Incluso encontramos distribuciones con uno o dos disquetes que se cargan en 2 MB de memoria. En máquinas un poco más recientes (de 32 a 128 MB de memoria) las distribuciones Toutou Linux o Puppy Linux lo hacen a las mil maravillas.

Sin embargo, no espere trabajar correctamente con una versión moderna de Linux y su entorno ofimático gráfico en estas condiciones pseudo-prehistóricas. Se deben respetar los requisitos siguientes:

- **Un procesador** (o más) de tipo Intel Pentium o superior o uno equivalente de la marca AMD. Atención: algunas distribuciones se compilan para funcionar únicamente a partir de Pentium o de Pentium II en adelante.
- **Al menos** 128 MB de memoria, pero 256 MB o más aportan una comodidad real de uso. Piense más bien en disponer de 512 MB, incluso 1 GB para obtener el mejor rendimiento. Dado el precio de la memoria, no es un lujo. Parte del principio que cuanta más memoria, mejor. Los PC recientes con 2 GB o 4 GB de memoria son ideales. En el marco de una instalación mínima en modo texto, 64 MB bastan.
- 500 MB de espacio en disco para una instalación mínima (sin interfaz gráfica y sólo con las herramientas básicas), pero de 2,5 GB a 4 GB para una instalación estándar, a la cual hay que añadir el espacio para los datos del usuario y la partición de intercambio.
- Una tarjeta gráfica (incluso antigua) compatible con la norma Vesa, que acepte una resolución de

1024 x 768 en 65 356 colores para el entorno gráfico. La resolución no tienen ninguna importancia si trabaja en modo texto.

---

 Son requisitos básicos. Si la frecuencia de reloj de su procesador es vital para la velocidad de ejecución de sus aplicaciones, ésta puede verse muy restringida por la falta de memoria o un disco duro demasiado lento. La cantidad de memoria física es un factor importante en el rendimiento. Cuanta más tengamos, mejor. Varios programas podrían funcionar paralelamente, la partición de intercambio no será necesaria y el sistema podrá utilizar más memoria caché para acelerar los accesos a los discos y periféricos. Si dispone de 512 MB o menos, considere ampliar a 1 GB o más. La diferencia es muy importante.

---

Las prestaciones globales siguen siendo aceptables en un Pentium II 300 con 256 MB para un uso ofimático o sencillo de Internet. Pero el sistema se resiente cuando varios programas se ejecutan al mismo tiempo. En un simple AMD Duron 800 con 512 MB, las prestaciones son excelentes para la mayoría de los usos habituales.

## 2. Compatibilidad del hardware

Antes de instalar Linux, compruebe si su hardware es compatible con Linux. Establezca una lista de los componentes de su ordenador y de sus diferentes periféricos. Los principiantes suelen alegar un problema con el soporte del hardware cuando falla la instalación. Una tarjeta gráfica, una impresora, un escáner determinados no funcionan correctamente o no del todo. Aunque Linux soporte la mayoría de los componentes de los ordenadores recientes, la compra de un ordenador moderno no es una garantía de buen funcionamiento.

En la lista, lo más importante no es la marca o el nombre del modelo comercial, sino el componente, el chip principal, llamado *chipset*, del producto. En el caso del Wi-Fi, poco importa que la tarjeta sea una Palmnet BZ46G. Pero si sabe que está construida a partir de un chip Centrino (Intel 2200 por ejemplo), entonces encontrará que funciona con Linux. Se deben evitar los productos de algunos fabricantes porque sus componentes no disponen de drivers que permitan utilizarlos. La casi totalidad del hardware de impresión propuesto por Hewlett-Packard funciona perfectamente con Linux. Por el contrario, hay que huir de las impresoras de tinta Lexmark (cuidado: no es la calidad del producto lo que está en entredicho, sino su soporte bajo Linux).

Si ya está instalado, el sistema operativo Windows puede ayudarle a inventariar su hardware. En el Panel de control puede acceder a Sistema y a la pestaña **Hardware**. Ya sólo le falta comprobar mediante un motor de búsqueda o sitios web especializados si sus componentes son compatibles.

A menos de que disponga de una máquina muy antigua, todas tarjetas gráficas funcionan. En todos los casos, Linux propone un driver genérico llamado «vesa» que, si no ofrece las mejores prestaciones, al menos permite utilizar todas las tarjetas compatibles con este estándar viejo de más de diez años. Algunos fabricantes ofrecen drivers muy eficaces. Las últimas tarjetas de los constructores Nvidia y AMD (ATI) están soportadas con drivers 3D que ofrecen las mismas prestaciones que en otros sistemas operativos. El sistema gráfico de Linux soporta por defecto un gran número de tarjetas, incluso con la aceleración 3D. Las mejores tarjetas gráficas han sido durante mucho tiempo las basadas en componentes NVIDIA mediante el driver propietario e Intel mediante el driver libre. Estos últimos años, gracias en gran medida a la ingeniería inversa en tarjetas NVIDIA (driver nouveau) y Radeon (driver Radeon), se han desarrollado drivers 3D libres. Su rendimiento ha alcanzado cotas comparables a los drivers propietarios.

Las tarjetas de sonido integradas en la placa base respetan un estándar de facto (AC97) soportado por Linux. Las tarjetas de sonido integradas en la placa base son rara vez componentes de alta gama. Una simple tarjeta Live que cuesta menos de 30 euros es mucho más eficaz. Algunos modelos específicos de tarjetas de sonido pueden crear problemas.

El Wi-Fi (con USB, PCCard, PCI, PCIe) debería funcionar o con un driver nativo para su hardware, o con la ayuda de una herramienta llamada Ndiswrapper que permite utilizar los drivers Windows para Linux. Según la versión que elija, puede ocurrir que no se le suministren pequeños componentes llamados firmwares, necesarios para la tarjeta Wi-Fi, y se deben adquirir por separado, ya sea desde el sistema de actualización o

desde un soporte (el famoso add-on) adicional, o por el constructor de la tarjeta. Los chipsets de marca Ralink, Broadcom o Intel son generalmente compatibles. El Bluetooth es perfectamente reconocido y soportado.

Varios sitios disponen de bases de datos de hardware compatibles para informarle con precisión. Los motores de búsqueda siguen siendo su mejor fuente. A título orientativo, le presentamos una lista de los sitios que le ayudarán en sus búsquedas:

- Lista de compatibilidad Novell y OpenSUSE: [http://en.opensuse.org/Hardware?LANG=en\\_UK](http://en.opensuse.org/Hardware?LANG=en_UK)
- Impresoras: <http://www.linuxfoundation.org/collaborate/workgroups/openprinting>
- Escáneres: <http://sane-project.org/>
- Periféricos USB en general: <http://www.qbik.ch/usb/devices/>
- Tarjetas de sonido: <http://www.alsa-project.org/>
- Tarjetas Wi-Fi: [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Linux/](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/)
- Módems internos o externos de tipo Windomem: <http://linmodems.org/>

---

 A pesar de toda la buena voluntad del mundo, ocurre a veces que un dispositivo no funcione en absoluto con Linux. ¿Quién tiene la culpa? Los drivers de periféricos suelen ser escritos a menudo por desarrolladores que ni siquiera tienen acceso a las especificaciones del dispositivo y que lo hacen todo por ingeniería inversa, o sea, intentando reproducir el funcionamiento del periférico desde su resultado. Requiere mucho tiempo. Algunos fabricantes respetan las reglas. A falta de proporcionar un verdadero driver, publican para algunos desarrolladores una documentación técnica. Otros proporcionan ellos mismos un driver con código cerrado para una distribución dada o que debe adaptar usted mismo a cada nueva versión de Linux. En este caso, no se garantiza el driver (esto ocurrió, por ejemplo, con las tarjetas basadas en chipset gráfico Kyro II o powerVR). Por lo tanto, antes de criticar a Linux si no le funciona un dispositivo, critique primero al fabricante de este último.

---

# Elegir una distribución

## 1. Debian



El proyecto Debian fue creado por Ian Murdock en 1993, en una época en la cual la idea misma de distribución Linux estaba todavía en pañales. El nombre de Debian proviene de Debra (la esposa de Murdock) e Ian. Durante mucho tiempo, Debian ha sido la única distribución entera y únicamente compuesta de programas libres y Open Source, lo que le sigue valiendo el nombre oficial de Debian GNU/Linux. Debian ha sido considerada durante algún tiempo la distribución Linux de referencia para la FSF. Las ventajas de Debian son numerosas:

- un número gigantesco de paquetes (se cifran en miles),
- un programa de instalación de software llamado APT muy práctico y eficaz,
- una distribución 100% open source,
- una estabilidad a toda prueba para un entorno de producción.
- muchos repositorios de programas

Estas ventajas conllevan también unos inconvenientes:

- paquetes a menudo antiguos,
- actualizaciones de la distribución irregulares y demasiado espaciadas,
- riesgos relacionados con la multiplicación de los paquetes y de los elementos dependientes,
- una instalación y una configuración complicadas.

---

 Todos estos inconvenientes no tienen por qué ser defectos. ¿Tenemos que preferir una antigua versión exenta de bugs o la última versión de un producto cuya fiabilidad no ha sido totalmente probada?

---

Todos estos elementos convierten a Debian en una distribución ideal para los informáticos, los ingenieros y los administradores de sistemas y redes, los entornos de producción en empresa, los puristas del software libre, los aficionados iluminados que no temen tirarse a la piscina. En cuanto a los principiantes, pasarán un pequeño calvario si se proponen aprender de forma autodidacta.

### a. Ubuntu



El multimillonario sudafricano Mark Shuttleworth, esencialmente conocido en mundo entero por haber sido uno de los primeros turistas en el espacio, pero también por haber hecho fortuna vendiendo su empresa Thawte, especializada en la seguridad, a Verisign, es un verdadero amante de la informática que ha contribuido como ningún otro al proyecto Debian. Para solventar algunos problemas de éste, creó la distribución Ubuntu Linux en 2005, con un presupuesto de 10 millones de dólares en contratación de desarrolladores. La palabra «Ubuntu» es un término del lenguaje africano bantú que significa «humanidad para los demás» o «soy lo que soy gracias a lo que somos todos». Este leitmotiv ejemplifica lo que la distribución se ha propuesto: ser un derivado de Debian, pero proporcionando los programas más recientes y orientados a la compatibilidad y la ergonomía. Ubuntu ofrece:

- una distribución que proviene de Debian,
- compatibilidad con los paquetes de Debian,
- un sistema de instalación muy simple,
- una salida cada 6 u 8 meses,
- un entorno gráfico agradable.

Esta distribución es ideal para los novatos y para los estudiantes. Se ha creado para ser la más fácil de instalar y de utilizar. Su éxito ha sido colosal y es la distribución más descargada e instalada desde hace varios años. Aunque a veces criticada por sus decisiones, Ubuntu es muy innovadora. El número de repositorios y su paquetería son impresionantes. En las secciones de descarga de muchos proyectos libres, se ofrece casi siempre un paquete Ubuntu. Está disponible en versiones Desktop y Servidor, con un largo periodo de soporte.

## b. Red Hat y Fedora



*Logo Red Hat*

Si existe una empresa comercial en el mundo de Linux que influyó y sigue marcando época, es la empresa Red Hat. Fundada en 1995 por Robert Young y Marc Ewing, edita la famosa distribución epónima cuya primera versión oficial data de 1994 (la empresa se fundó tras el lanzamiento de la distribución). El sistema de paquete RPM apareció con la versión 2.0. Fue tal la buena acogida de Red Hat que lleva casi diez años siendo la referencia. Cada versión era innovadora tanto en la integración de los programas como en su instalador (llamado «anaconda») y sus herramientas de configuración.

En el año 2003, Red Hat publicó la versión 9.0, la última destinada oficialmente al gran público. Se confió las siguientes versiones al proyecto comunitario **Fedora**, que sigue sacando cada seis meses una nueva versión. Red Hat se concentra ahora en el mundo de la empresa con distribuciones comerciales llamadas**RHEL** (*Red Hat Enterprise Linux*) y cuyas características son:

- versiones profesionales destinadas a las empresas,
- soluciones que van del puesto de trabajo al servidor más potente,
- arquitecturas físicas numerosas,
- un soporte comercial,
- actualizaciones garantizadas durante siete años,
- 100% libre.

Aunque es posible instalar una versión RHEL AS (*Advanced Server*) en un PC de oficina, se puede dudar del interés que tiene para un puesto de trabajo o un principiante. Aunque libre (puede disponer de sus ficheros fuente libremente), el coste del soporte es muy elevado. No obstante, si no le da miedo su instalación, la distribución **CentOS** (*Community Enterprise Operating System*) es una copia exacta y descargable de RHEL en la que se ha suprimido cualquier rastro de los nombres y logos de Red Hat. Red Hat o CentOS serán la elección del iniciado que desea aprender directamente con una distribución Linux utilizada en empresa.



Logo Fedora

En cuanto al proyecto Fedora, sigue un ciclo de desarrollo rápido y continua destinado al gran público. Su instalación es sencilla. Sin embargo, al conjunto le falta un poco de coherencia (por ejemplo, la herramienta de partición de los discos no está accesible durante la instalación), lo que lo convierte en una distribución ideal para todos aquellos, aficionados iluminados que desean entrar un poco más en los pequeños detalles.

### c. Mandriva (ex-Mandrake)



Mandriva Linux (ex-Mandrake) es una distribución derivada de la distribución Red Hat y durante mucho tiempo completamente compatible con ella. Fue creada por Gaël Duval con el fin de integrar a la distribución el entorno gráfico de oficina KDE, al contrario que el de Red Hat, que integraba el entorno GNOME. Durante varios años, Mandrake fue la distribución estrella en fuerte competición con Red Hat. Mandrake era de hecho (y sigue siendo) más amena. Su proceso de instalación es un modelo a seguir y su utilización resulta de las más sencillas. Renombrada como Mandriva tras la adquisición por la empresa Connectiva, la distribución pierde influencia a partir de entonces. Las razones son múltiples, pero muy relacionadas con los avatares de la empresa Mandriva. Una gestión difícil y una mala orientación en los años 2000-2001 (el e-learning y la experiencia estadounidense de Start-up) estuvo a punto de llevarla a su bancarrota una primera vez y, tras un concurso judicial del que consiguió salir, se encontró de nuevo con problemas poco después. La introducción en el mercado bursátil no dio los resultados esperados. Esta distribución es poco atractiva para

los profesionales, que la consideran orientada al gran público. Sus soluciones, que satisfacen al gran público, siguen siendo consideradas de las mejores técnicamente, pero sufre algunos problemas de estabilidad.

Mandriva, no en vano, sigue innovando constantemente, en particular en el puesto de trabajo nómada con versiones que pueden ser cargadas desde un USB. Esta distribución se beneficia del boca a boca entre los principiantes.

#### d. openSUSE

Se pronuncia *opensus*; **OpenSUSE** es una distribución de origen alemán del año 1992. El nombre de la empresa era de por sí un homenaje al famoso **Konrad Zuse**, el inventor de los ordenadores modernos. La distribución está basada originalmente en Slackware. En 1996, SuSE se acerca a una distribución francesa llamada **Jurix**, creada por Florian La Roche, que se utiliza como base en lugar de Slackware. Este mismo año arranca el desarrollo de la herramienta YaST y sale la versión 4.2, en realidad totalmente nueva. Al mismo tiempo, SuSE emplea el nuevo gestor de paquetes de Red Hat, llamado RPM.

A principios de 1997, SuSE se embarca en la aventura americana al instalar nuevas oficinas en Oakland. Entre 1997 y 2003, la distribución SuSE no deja de ser mejorada para convertirse en una referencia en cuestión de sencillez de instalación, administración y uso.

Hoy en día, el futuro de la distribución está garantizado; la empresa Novell compró primero la empresa Ximian, especializada en el desarrollo Open Source de herramientas para Linux, entre las que destacan un completo escritorio Gnome, un sistema de mensajería llamado Evolution y un asistente de configuración llamado Red Carpet. Novell anuncia la compra de la empresa SuSE en enero de 2004. El desarrollo pasa ahora a ser comunitario, bajo el proyecto **openSUSE**. Cualquiera que lo desee puede participar en la mejora del producto. En contraprestación, Novell se compromete a proporcionar a la comunidad cada seis a ocho meses una versión estable, libre y gratuita. La reciente fragmentación de Novell en varias estructuras no ha disminuido la vida de la distribución, que sirve de base a la distribución SLES (*Suse Linux Enterprise Server*) de Novell, el gran competidor de Red Hat.



Gecko, mascota de openSUSE

#### e. Las demás

Es imposible nombrar todas las distribuciones, dado las muchas que hay. Además de las grandes distribuciones que acabamos de reseñar, se debe hacer mención a otros nombres. La distribución **Slackware** es una de las más antiguas. Incluso se entregaba en un disquete. Durante los primeros años de vida de Linux, la Slackware era la distribución de referencia para aprender a utilizar Linux. Es extremadamente austera: su instalador se reduce a la mínima expresión y casi todas las configuraciones deben establecerse a mano, sin asistente. No cuenta con un gestor de paquetes (se trata de simples archivos de ficheros comprimidos). Todo ello hace que sea ideal para los manitas y los apasionados de Unix,

pero no tanto para los principiantes.

La distribución **Gentoo** es muy peculiar. En vez de entregarle todos los programas ya listos para su uso, su instalador va a determinar exactamente con su ayuda la configuración de su máquina, y en particular su modelo de procesador. En función de ello, compilará (transformará el programa fuente en forma de lenguaje comprensible en lenguaje máquina) cada componente, que habrá seleccionado con todas las optimizaciones previstas para su hardware. Eso se llama una distribución fuente. El resultado puede ser interesante: las prestaciones de sus programas aumentan, siendo de media del 10 al 20% más potentes. ¡Pero a qué coste! La instalación no es ni de lejos fácil para los principiantes y, sobre todo, es muy larga: varias horas (incluso decenas de horas) según los programas que quiera y la potencia de su máquina.

Otra distribución sorprendente es la **LFS** (*Linux From Scratch*). No es precisamente una distribución, sino más bien una guía que le proporciona un método para construir su propia configuración. Paso a paso, le corresponde elegir sus diversos componentes y la configuración de su sistema. Así, se asegura de que obtiene exactamente la distribución que desea, ni más o menos. Pero también en este caso, los principiantes, e incluso de hecho los aficionados iluminados, lo evitan.

Al margen de todas estas distribuciones, encontramos numerosos derivados. **Aurox Linux** deriva de Red Hat. **PCLinuxonline** deriva de Mandriva. **Kunbuntu** deriva de Ubuntu (o más bien es una distribución Ubuntu que integra el entorno de escritorio KDE), que deriva de Debian. **CentOS** deriva de RHEL, **Mint** deriva de Ubuntu, y así sucesivamente. Además, existen también las minidistribuciones que caben en un minicd o una llave USB, y es lo ideal para arreglar un ordenador.

Puede encontrar una lista exhaustiva de distribuciones Linux en distrowatch: <http://distrowatch.com/>

## 2. Los LiveCD

El LiveCD es una categoría sorprendente. Seguramente usted habrá probado Linux para ver qué se parece o para comprobar si funciona correctamente con su hardware. En vez de instalarlo en su disco duro (si esta etapa le asusta, el capítulo El Shell y los comandos GNU de este libro le propone una guía paso a paso para instalar su Linux), piense primero en probar Linux sin instalarlo. El LiveCD sirve principalmente para eso: es una instalación completa de Linux muy comprimida y que cabe en un solo CD o DVD (en este caso, se habla del liveDVD).

Utilizar un LiveCD es muy sencillo: inserte el CD o DVD en su lector y vuelva a encender su ordenador. Después de haber comprobado la configuración de su máquina (el setup del BIOS, véase las instrucciones de uso de su ordenador), su lector de CD o DVD debe ser el primer dispositivo en cargarse. Después de unos segundos (o a veces minutos) de carga en memoria, aparece el escritorio y todos los programas más conocidos. No se instala nada en su disco duro. El LiveCD más famoso actualmente es **Knoppix**. Está basado en una distribución Debian y, además, cuenta con un mecanismo para copiarlo en el disco duro. Cada nueva versión de la mayoría de las distribuciones (openSUSE, Fedora, Ubuntu, Mageia, etc.) llega con un LiveCD para probar las últimas novedades sin instalarla.

Una última categoría es la de los discos de recuperación. Ultimate Boot CD o System Rescue CD ofrecen varias herramientas que usan Linux para reparar su sistema. Cabe destacar que con estas herramientas es muy fácil desde Linux averiguar una contraseña Windows si la ha perdido.

# Obtener ayuda

## 1. La ayuda propia de los comandos

No es posible conocer de memoria todos los parámetros y argumentos de un comando. Por fortuna, Linux propone como mínimo dos mecanismos para conocer la sintaxis de un comando. La mayor parte del tiempo, el parámetro `--help` muestra la ayuda incluida directamente dentro del programa al que se llama. A veces la ayuda aportada es suficiente para encontrar lo que está buscando. Es lo que sucede con el comando `date`, cuya salida está voluntariamente cortada aquí porque, si no, necesitaría dos páginas.

```
$ date --help
Uso: fecha [OPTION]... [+FORMAT]
      o: fecha [-u|--utc|--universal] [MMDDhhmm[[CC]AA][.ss]]
Mostrar la fecha corriente según el FORMAT especificado o
 inicializar la fecha del sistema.

-d, --date=CADENA      mostrar la fecha según la descripción dada
                        por la CADENA,
                        excluyendo la palabra reservada « now »
-f, --file=FICHERO    idéntico a --date para cada línea del
                        FICHERO de fechas
-r, --reference=FILE   display the last modification time of FILE
-R, --rfc-2822         output date and time in RFC 2822 format.
...
...
```

Puede ocurrir que la ayuda sea demasiado concisa o le falten explicaciones, o bien que esté totalmente ausente. En este caso, se considera `--help` como un parámetro no válido y puede que obtenga un mensaje de error o una línea de información:

```
$ cal --help
cal: opción inválida -- -
uso: cal [-13smjyV] [[mes] año]
```

La última línea no explica la sintaxis de los parámetros.

## 2. La ayuda interna del shell

Los comandos internos no aceptan el parámetro `--help`, pero para estos comandos el intérprete de comandos propone un comando `help`. Si se utiliza de forma aislada, proporciona la lista de los comandos internos. Si le proporciona como parámetro el nombre de un comando interno, se muestra la ayuda correspondiente. De esta manera, puede aprender que `pwd` admite dos parámetros opcionales.

```
$ help pwd
pwd: pwd [-LP]
      Print the current working directory. With the -P option, pwd
      prints the physical directory, without any symbolic links; the -L
      option makes pwd follow symbolic links.
```

## 3. El manual en línea

### a. Acceso

Cuando los dos mecanismos de ayuda anteriores resultan ser insuficientes, es probable que la ayuda

buscada se encuentre en el manual de Unix. Esta manual es estándar en todos los Unix, incluso Linux, y no importa cuál sea el shell, ya que se trata de un comando externo.

El manual está accesible desde el comando **man**. Puede hacer una sencilla prueba con el comando **date**:

```
$ man date
```

El modo de empleo del comando que se pasa como parámetro de man se visualiza.

## b. Estructura de una página

The screenshot shows a terminal window titled "Sesión de Screen - Konsole". The window contains the man page for the "date" command. The title bar has icons for minimize, maximize, and close. The menu bar includes "Sesión", "Editar", "Vista", "Marcadores", "Preferencias", and "Ayuda". The main area displays the man page content:

```
DATE(1) User Commands DATE(1)

NAME
    date - print or set the system date and time

SYNOPSIS
    date [OPTION]... [+FORMAT]
    date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]

DESCRIPTION
    Display the current time in the given FORMAT, or set the system date.
    -d, --date=STRING
        display time described by STRING, not 'now'
    -f, --file=DATEFILE
        like --date once for each line of DATEFILE
    -r, --reference=FILE
        display the last modification time of FILE
    -R, --rfc-2822
        output date and time in RFC 2822 format. Example: Mon, 07 Aug
        2006 12:34:56 -0600
    --rfc-3339=TIMESPEC
        output date and time in RFC 3339 format. TIMESPEC='date', 'sec-
        onds', or 'ns' for date and time to the indicated precision.
        Date and time components are separated by a single space:
Manual page date(1) line 1
```

The status bar at the bottom shows icons for "Sesión de Screen" and "Terminal".

*El manual en línea*

Una página de manual se compone de varias secciones, entre las cuales están las siguientes, aunque no se hallan obligatoriamente todas presentes:

- **Nombre:** nombre y papel del comando.
- **Sinopsis:** sintaxis general, parámetros y argumentos aceptados.
- **Descripción:** modo de empleo detallado del funcionamiento del comando y de los argumentos principales.
- **Opciones:** descripción detallada de cada parámetro posible, en general en forma de lista.
- **Ejemplos:** el manual puede proporcionar ejemplos concretos de uso del comando.
- **Entorno:** el comando puede funcionar de manera diferente dependiendo de los valores que adopten algunas de las variables del shell.

- **Conformidad:** el comando está conforme a unas recomendaciones o normas (por ejemplo, POSIX).
- **Errores (bugs):** el comando puede a veces funcionar mal en casos puntuales que se pueden enumerar en este sitio.
- **Diagnóstic/retorno:** el comando, según su resultado, puede devolver códigos de errores significativos cuyo valor permite determinar el tipo de problema (fichero con argumento ausente, etc.).
- **Ver también:** lista de los comandos relacionados con el programa que pueden interesar al usuario.

### c. Navegación

Se navega muy fácilmente por la ayuda:

- La barra de espacio desplaza una página completa.
- La tecla [Intro] desplaza línea por línea.
- Las teclas [Arriba] y [Abajo] desplazan una línea arriba o abajo.
- Las teclas [Re Pág] y [Av Pág] desplazan media página arriba o abajo.
- Las teclas [Principio] y [Fin] hacen exactamente lo que se espera de ellas.
- La tecla / permite una búsqueda. /toto busca toto. En este caso, la tecla n busca la coincidencia siguiente, mientras que [Mayús] n (N) busca la anterior.
- La tecla Q sale de la ayuda y vuelve al shell.

### d. Las secciones

El manual de Linux no sólo hace referencia a los comandos clásicos. Es un manual mucho más completo. Los comandos simples, los de administración, los ficheros de configuración, los periféricos, las llamadas al sistema, las funciones de programación de varios lenguajes y mucho más pueden tener reflejo en estos manuales. Por eso, el manual se compone de varias secciones.

Sección	Contenido
1	Instrucciones ejecutables o comandos del shell
2	Llamadas del sistema (API del núcleo...)
3	Llamadas de las librerías (funciones C...)
4	Ficheros especiales (contenido de /dev como sd, hd, pts, etc.)
5	Formato de los ficheros (/etc/passwd, /etc/hosts, etc.)
6	Juegos, salvapantallas, programas varios, etc.
7	Varios, comandos no estándares que no encuentran sitio en otra parte
8	Comandos de administración del sistema Linux

Puede ocurrir que a veces la llamada al manual para un comando no devuelva la página del manual correspondiente. Lo que pasa es que man busca por defecto la primera coincidencia en el orden de las secciones. Si usted busca ayuda relativa al formato del fichero de las contraseñas, llegará primero a la ayuda del comando **passwd**. Mire el encabezamiento de la página. El número de la sección está indicado justo detrás del nombre del comando, entre paréntesis. El comando **man** ha encontrado una coincidencia de **passwd** en la sección 1 y muestra la página del manual asociada.

```
$ man passwd
PASSWD(1)           Manual del usuario Linux
NOMBRE
passwd - actualizar las marcas de autenticación de un usuario.
...
```

Puede pedir a man que busque el manual correspondiente en una sección específica indicando su número justo antes del nombre del comando. Para acceder al manual del fichero **passwd**, haga como a continuación:

```
$ man 5 passwd
PASSWD(5)           Manual del administrador Linux
NOMBRE
passwd - Fichero de contraseñas.
...
```

### e. Buscar por correspondencia

Si duda del comando que debe utilizar o no recuerda su nombre, o incluso si quiere conocer todos los comandos relacionados con una palabra, entonces utilice el parámetro **-k** de man:

```
$ man -k passwd
/etc/rpasswd.conf (5) [rpasswd.conf] - configuration file for remote
password update client
chpasswd (8)          - change user passwords in batch
Crypt::SmbHash (3pm) - Perl-only implementation of lanman and nt md4
hash functions, for use in Samba style smbpasswd entries
fgetpwent_r (3)       - get passwd file entry reentrantly
getpwent_r (3)       - get passwd file entry reentrantly
gpasswd (1)          - change group password
ldappasswd (1)        - change the password of an LDAP entry
lppasswd (1)          - add, change, or delete digest passwords.
makepasswd (1)         - generate and/or encrypt passwords
mkpasswd (1)           - Overfeatured front end to crypt(3)
pam_localuser (8)      - require users to be listed in /etc/passwd
pam_rpasswd (8)        - PAM module to change remote password
passwd (1)             - change user password
passwd (1ssl)          - compute password hashes
passwd (5)             - password file
passwd2des (3)         - RFS password encryption
rpasswd (1)            - change user password on remote server
rpasswd.conf (5)        - configuration file for remote password update client
rpasswdd (8)           - remote password update daemon
saslpasswd2 (8)         - set a user's sasl password
smbpasswd (5)           - The Samba encrypted password file
smbpasswd (8)           - change a user's SMB password
vncpasswd (1)           - set passwords for VNC server
yppasswd (1)            - change your password in the NIS database
```

## 4. Buscar ayuda en Internet

Como ya se ha indicado al principio de este capítulo, hay toda una comunidad alrededor de Linux y del software libre, y los editores de distribuciones proporcionan la documentación y el soporte. Por eso, se dispone de muchos medios para obtener ayuda, en particular en Internet:

- la documentación del editor,
- los sitios comunitarios (FAQ, foros),
- los grupos de noticias,
- el proyecto de documentación libre (HOWTOs),
- etc.

Piense primero en la documentación de los editores:

- Red Hat: <http://www.redhat.com/support>
- Debian: <http://www.debian.org/doc/>
- openSUSE: <http://en.opensuse.org/Documentation>
- Ubuntu: <https://help.ubuntu.com/>
- Mandriva: <http://club.mandriva.com/xwiki/bin/view/KB/OfficialDocumentation>
- Mageia: <http://www.mageia.org/es/support/>
- Fedora: <http://docs.fedoraproject.org>

En cada uno de estos sitios, encontrará también probablemente:

- una base de conocimiento,
- un Wiki,
- un foro,
- informes de bugs.

Resulta imposible relacionar todos los sitios de la comunidad; pero a continuación encontrará algunos:

- LinuxEs: <http://www.linux.es/>
- Freshmeat: <http://freshmeat.net>
- Slashdot: <http://slashdot.org>
- Foro Linux: <http://www.linuxespanol.com>
- Foro Fedora: <http://fedora-es.com/>
- Foro Mandriva: <http://forum.mandriva.com/index.php?op=Es>
- Foro Debian: <http://es.debian.org/foro>
- Foro Ubuntu: <http://ubuntu-es.org/index.php?q=forum>
- Foro openSUSE: Alionet: [http://es.opensuse.org/punto\\_de\\_encuentro](http://es.opensuse.org/punto_de_encuentro)

Entre los sitios de documentación:

- Lea Linux: <http://lea-linux.org>
- The Linux Documentation Project: <http://tldp.org>

- LinuxDocs: <http://linuxdocs.org>

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

## **1. Preguntas**

### **El sistema operativo Unix**

**1** ¿Cuál es el único lenguaje que puede comprender directamente un ordenador?

**2** ¿Cuál es, en resumen, el papel de un sistema operativo?

**3** ¿Forma parte del sistema operativo UNIX la interfaz gráfica?

**4** ¿Qué significa multitarea?

- A - Que un programa puede efectuar varias tareas al mismo tiempo.
- B - Que el sistema sabe iniciar varios programas unos tras otros.
- C - Que el ordenador sabe hacer varias cosas.
- D - Que el sistema operativo sabe ejecutar varios programas de forma simultánea.

**5** En un sistema multiusuario:

- A - Varias personas pueden conectarse al mismo tiempo.
- B - Los programas pueden ejecutarse con nombres de usuarios diferentes.
- C - Un usuario debe estar conectado para ejecutar programas.
- D - Varios usuarios pueden conectarse al mismo tiempo.

**6** ¿Para qué sirven las llamadas al sistema de un núcleo Unix?

**7** ¿De qué forma la gestión de las entradas y salidas permite acceder a los periféricos?

**8** ¿Qué sistema operativo está en el origen de Unix?

- A - Ultrix.
- B - Minix.
- C - Multics.
- D - Xenix.

**9** Nombre al menos dos personas que hayan contribuido a la creación de Unix.

**10** ¿Qué significa UNIX?

**11** ¿En qué máquina se desarrolló UNIX en primer lugar?

**12** ¿Por qué se dice que Unix es portable?

- A - Porque su código fuente es de pequeño tamaño.
- B - Porque el sistema funciona en ordenadores portátiles.
- C - Porque su código fuente está mayoritariamente escrito en lenguaje C.
- D - Unix no es portable.

**13** ¿Qué significa BSD?

**14** ¿Cuál es la empresa que escribió primero Unix?

**15** ¿Cuáles son los dos principales estándares Unix?

**16** Nombre tres Unix propietarios.

**17** ¿Cuál ha sido el primer Unix que ha funcionado en un PC?

**18** ¿Cuál es el papel de The Open Group?

**19** ¿Es Unix un estándar?

### **El software libre y Linux**

**20** ¿Qué significa FSF?

- A - Free Society for Freedom.
- B - Freeware Survey Front.
- C - Free Software Foundation.
- D - Foundation for Software to be Free.

**21** ¿Quién es Richard Stallman?

**22** ¿Puede un programa libre ser de pago?

**23** ¿Es legal copiar y dar una distribución de tipo Fedora a sus conocidos?

**24** ¿En qué año empezó el desarrollo de Linux?

**25** ¿Quién empezó a escribir Linux?

**26** ¿En qué año salió la primera versión de Linux?

- A - 1991
- B - 1992
- C - 1993
- D - 1994

**27** Nombre tres distribuciones de Linux conocidas.

**28** ¿A partir de qué modelo de procesador Intel funciona Linux?

**29** ¿Su Quad Core de 64 bits está soportado por Linux?

**30** ¿A partir de qué cantidad de memoria se puede hacer funcionar Linux de manera cómoda y su entorno gráfico de manera óptima?

**31** ¿Cómo saber si su hardware es compatible con Linux?

**32** Su impresora es una Lexmark 7500. ¿Dónde encontrará si existe un driver compatible con Linux?

**33** ¿Cuál es la principal diferencia entre una distribución Debian y una distribución Red Hat?

**34** Como muy principiante, ¿qué distribuciones le vienen mejor?

- A - Fedora
- B - Ubuntu
- C - openSUSE
- D - Debian

**35** ¿Por qué, para una empresa, podría ser más tranquilizador comprar licencias de distribuciones comerciales como Red Hat o Novell?

**36** ¿Cómo probar Linux sin instalarlo?

### **Acceso a la ayuda**

**37** Desconoce la sintaxis del comando **who**. Para obtener ayuda desde este comando, tiene que teclear:

- A - who --help
- B - who -h
- C - who help
- D - man who

**38** No recuerda el nombre exacto de un comando; sólo que contiene la expresión «passwd». Para obtener la lista del conjunto de las páginas de los manuales correspondiente a esta palabra, ¿qué comando debe teclear?

**39** Al teclear «man passwd» encuentra el manual del comando **passwd**. ¿Qué comando permite acceder al manual que describe la sintaxis del fichero **/etc/passwd**?

**40** ¿En qué sitio de Internet en español puede obtener información de última hora sobre Linux y los programas libres?

## 2. Resultados

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /40

Para este capítulo, su resultado mínimo debe ser de 30 respuestas acertadas (30/40).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- El sistema operativo Unix.
- El software libre y Linux.
- Acceso a la ayuda.

## 3. Respuestas

### El sistema operativo Unix

**1** ¿Cuál es el único lenguaje que puede comprender directamente un ordenador?

*El lenguaje binario, que es el propio de cada microprocesador.*

**2** ¿Cuál es, en resumen, el papel de un sistema operativo?

*Un sistema operativo es la interfaz entre el hardware y los programas.*

**3** ¿Forma parte del sistema operativo UNIX la interfaz gráfica?

*No. El entorno gráfico no está incluido en el sistema. Es un producto complementario que funciona en el espacio de trabajo del usuario.*

**4** ¿Qué significa multitarea?

- A - Que un programa puede efectuar varias tareas al mismo tiempo.
- B - Que el sistema sabe iniciar varios programas unos tras otros.
- C - Que el ordenador sabe hacer varias cosas.

- D - Que el sistema operativo sabe ejecutar varios programas de forma simultánea.

D.

**5** En un sistema multiusuario:

- A - Varias personas pueden conectarse al mismo tiempo.
- B - Los programas pueden ejecutarse con nombres de usuarios diferentes.
- C - Un usuario debe estar conectado para ejecutar programas.
- D - Varios usuarios pueden conectarse al mismo tiempo.

A y B

**6** ¿Para qué sirven las llamadas al sistema de un núcleo Unix?

*Son las funciones de programación del núcleo que los programas utilizan para acceder a sus recursos y a los del ordenador.*

**7** ¿De qué forma la gestión de las entradas y salidas permite acceder a los periféricos?

*Los periféricos son vistos como ficheros; usted accede a ellos a través de ficheros.*

**8** ¿Qué sistema operativo está en el origen de Unix?

- A - Ultrix.
- B - Minix.
- C - Multics.
- D - Xenix.

C

**9** Nombre al menos dos personas que hayan contribuido a la creación de Unix.

*Dos entre: Ken Thompson, Dennis Ritchie, Rudd Canaday, Brian Kernighan.*

**10** ¿Qué significa UNIX?

*Uniplexed Information and Computing System.*

**11** ¿En qué máquina se desarrolló UNIX en primer lugar?

*El DEC PDP-7.*

**12** ¿Por qué se dice que Unix es portable?

- A - Porque su código fuente es de pequeño tamaño.
- B - Porque el sistema funciona en ordenadores portátiles.
- C - Porque su código fuente está mayoritariamente escrito en lenguaje C.
- D - Unix no es portable.

C

**13** ¿Qué significa BSD?

*Berkeley Software Distribution.*

**14** ¿Cuál es la empresa que escribió primero Unix?

*AT&T.*

**15** ¿Cuáles son los dos principales estándares Unix?

*BSD y System V.*

**16** Nombre tres Unix propietarios.

*Tres entre: Solaris, AIX, HP-UX, True64, SCO, Qnx, etc.*

**17** ¿Cuál ha sido el primer Unix que ha funcionado en un PC?

*Xenix.*

**18** ¿Cuál es el papel de The Open Group?

*Es depositario de la marca Unix y de las recomendaciones de compatibilidad.*

**19** ¿Es Unix un estándar?

*Si, IEEE 1003 es un estándar conocido con el nombre de POSIX.*

### **El software libre y Linux**

**20** ¿Qué significa FSF?

- A - Free Society for Freedom.
- B - Freeware Survey Front.
- C - Free Software Foundation.
- D - Foundation for Software to be Free.

*C*

**21** ¿Quién es Richard Stallman?

*El presidente de la FSF y el fundador del movimiento de los programas libres.*

**22** ¿Puede un programa libre ser de pago?

*Sí, sólo las fuentes deben ser publicadas gratuitamente.*

**23** ¿Es legal copiar y dar una distribución de tipo Fedora a sus conocidos?

*Sí, la distribución es libre y las libertades ofrecidas por la licencia GPL le autoriza a ello.*

**24** ¿En qué año empezó el desarrollo de Linux?

*En 1991.*

**25** ¿Quién empezó a escribir Linux?

*Linus Torvalds.*

**26** ¿En qué año salió la primera versión de Linux?

- A - 1991
- B - 1992

- C - 1993
- D - 1994

D

**27** Nombre tres distribuciones de Linux conocidas.

*Red Hat, Debian, openSUSE, Fedora, etc.*

**28** ¿A partir de qué modelo de procesador Intel funciona Linux?

*De los modelos 80386 y siguientes.*

**29** ¿Su Quad Core de 64 bits está soportado por Linux?

*Sí, Linux soporta los procesadores de 64 bits y los multiprocesadores.*

**30** ¿A partir de qué cantidad de memoria se puede hacer funcionar Linux de manera cómoda y su entorno gráfico de manera óptima?

*256 MB pueden bastar, pero es preferible para el uso de KDE o Gnome disponer de al menos 512 MB.*

**31** ¿Cómo saber si su hardware es compatible con Linux?

*Dirigiéndose a las listas de compatibilidad del editor de la distribución, o a los sitios comunitarios, o incluso al sitio del fabricante.*

**32** Su impresora es una Lexmark 7500. ¿Dónde encontrará si existe un driver compatible con Linux?

*En el sitio <http://www.linuxprinting.org>.*

**33** ¿Cuál es la principal diferencia entre una distribución Debian y una distribución Red Hat?

*La distribución Debian es libre y comunitaria, mientras que la Red Hat, aunque libre, es comercial.*

**34** Como muy principiante, ¿qué distribuciones le vienen mejor?

- A - Fedora
- B - Ubuntu
- C - openSUSE
- D - Debian

*B y C, porque Debian es demasiado complejo para un principiante y Fedora gustaría más a los aficionados iluminados.*

**35** ¿Por qué, para una empresa, podría ser más tranquilizador comprar licencias de distribuciones comerciales como Red Hat o Novell?

*Por el soporte, el seguimiento de los parches de los errores y la seguridad.*

**36** ¿Cómo probar Linux sin instalarlo?

*Con un liveCD.*

### **Acceso a la ayuda**

**37** Desconoce la sintaxis del comando **who**. Para obtener ayuda desde este comando, tiene que teclear:

- A - who --help

- B - who -h
- C - who help
- D - man who

A

**38** No recuerda el nombre exacto de un comando; sólo que contiene la expresión «passwd». Para obtener la lista del conjunto de las páginas de los manuales correspondiente a esta palabra, ¿qué comando debe teclear?

*man -k passwd*

**39** Al teclear «man passwd» encuentra el manual del comando **passwd**. ¿Qué comando permite acceder al manual que describe la sintaxis del fichero **/etc/passwd**?

*El manual está en la sección 5: man 5 passwd.*

**40** ¿En qué sitio de Internet en español puede obtener información de última hora sobre Linux y los programas libres?

*<http://www.linux.es/>*

# Prácticas

## 1. Historia de Unix y de Linux

- Diríjase al sitio de Éric Lévénez: <http://www.levenez.com/>. Éric Lévénez mantiene al día la historia de todos los sistemas Unix desde su origen. Con la ayuda de los documentos «Unix History», determine:

- a - La fecha exacta de Unix V1.
- b - La fecha exacta de 1BSD.
- c - La fecha exacta de Linux 1.0.

*Respuestas:*

- a - Unix V1 salió el 3 de noviembre de 1971.
- b - 1BSD salió el 9 de marzo de 1978.
- c - Linux 1.0 salió el 14 de marzo de 1994.

- Le invitamos a que vaya al sitio de The Open Group. Intente encontrar los documentos que describen el último estándar Shell y Herramientas.

- a - ¿Cuáles son sus referencias?

- b - ¿Puede acceder a ellas gratuitamente?

- c - Segundo este documento, ¿cuáles son los únicos parámetros obligatorios del comando **du** para un Unix POSIX?

*Respuestas:*

- a - Lleva la referencia C048, se llama «Shell and Utilities, Issue 6, 2004 Edition [XCU]», accesible desde <http://www.opengroup.org/bookstore/catalog/c048.htm>. Es un verdadero estandar IEEE Std 1003.1, 2004 Edition.

- b - Puede acceder de manera gratuita a la versión HTML en línea del documento. Diríjase a <http://www.opengroup.org/onlinepubs/009695399/toc.htm> para leer este documento. Sin embargo, se le pedirán sus datos (apellido, nombre y mail).

- c - Segundo este documento, sólo las opciones **-a**, **-H**, **-k**, **-L**, **-s** y **-x** son obligatorias para cualquier Unix Posix.

## 2. Distribuciones

- El sitio Distrowatch <http://distrowatch.com/> proporciona información completa sobre todas las distribuciones Linux conocidas, así como estadísticas. Diríjase a este sitio y conteste a las preguntas siguientes:

- a - ¿Cuál es la especificidad de la distribución Gentoo?

- b - ¿Puede un principiante instalar la distribución LFS?

- c - Nombre, a fecha de lectura, las cinco distribuciones en cabeza de las estadísticas.

*Respuestas:*

- a - Teclee Gentoo en el campo situado arriba a la izquierda. La descripción le informa de que Gentoo dispone de una herramienta de supervisión particular llamada Portage. Si profundiza más, aprenderá que Gentoo acepta la compilación sobre la marcha de los paquetes para adaptarse perfectamente a su OS.

- b - LFS no está destinada a los principiantes. No es una distribución, sino un método para crear su propia distribución. Tiene que seguir una guía para crear su instalación Linux desde cero (Linux From Scratch).

- c - A fecha del 26/06/2008, las distribuciones más vistas son, por orden, Ubuntu, openSUSE, Fedora, Mint y PCLinuxOS. Se puede acceder a esta lista a la derecha, en las estadísticas de las páginas.

- Diríjase al sitio oficial de openSUSE <http://www.opensuse.org/> y conteste a las preguntas siguientes:

- a - ¿Cuáles son los soportes y métodos posibles para descargar openSUSE?

- b - ¿La distribución openSUSE puede soportar la tarjeta Creative Audigy2?

- c - ¿Qué es Compiz Fusion?

*Respuestas:*

- a - Haga clic en «Get It», en la página de inicio. Se puede descargar openSUSE en forma de LiveCD, DVD o de instalación vía red, para las arquitecturas de 32 bits, 64 bits y PowerPC. La

descarga se puede hacer vía BitTorrent o transferencia HTTP/FTP clásica. El conjunto es gratuito.

b - En el wiki del sitio, teclee Audigy. Se proponen dos vínculos (a fecha del 26/06). La palabra HCL significa Hardware Compatibility List. Haga clic en el vínculo correspondiente. Así aprenderá que todas las tarjetas de sonido Audigy son compatibles.

c - Compiz Fusion ([http://en.opensuse.org/Compiz\\_fusion](http://en.opensuse.org/Compiz_fusion)) es un gestor de ventanas composite que permite obtener numerosas funcionalidades avanzadas para el escritorio y efectos 3D a veces tan bonitos como inútiles. Se puede instalar con algunos clics bajo Linux.

### 3. Ayuda y documentación

1. Le hemos hablado del comando **dialog**, pero no entiende su funcionamiento. El objetivo es lograr aprovechar este comando gracias a su ayuda.
  - a - Acceda a la ayuda interna de dialog, en particular a la relativa a la visualización del calendario. Cree un calendario del mes de junio de 2009 con un texto en una línea «elija la fecha» y una anchura automática.
  - b - El título de la pantalla es «elección de la fecha» y el título del cuadro es «calendario». Pero la ayuda interna no está clara. Acceda al manual de dialog. ¿Cuáles son los dos parámetros que hay que pasar a dialog? Muestre el nuevo cuadro de diálogo.
  - c - Efectúe una búsqueda global de dialog en el manual. Si encuentra el comando **Xdialog**, acceda a su ayuda. ¿Cuál es la diferencia con dialog? Si puede, muestre el mismo calendario pero con Xdialog.

*Respuestas:*

a - La opción `--calendar` muestra un calendario. El primer parámetro es el texto «Seleccione la fecha». Los dos siguientes son la altura y la anchura. 0 indica un tamaño automático. Especifique luego la fecha con el formato «día mes año».

El título de la ventana es `--backtitle`, mientras que el título de la caja es `--title`.

Así, el comando es:

```
$ dialog --backtitle "Elección de la fecha" --title "Calendario"  
--calendar "Seleccione la fecha" 0 0 01 06 2009
```

b - Acceda a la ayuda con «man dialog». En las opciones comunes (Common Options), el parámetro `--backtitle` permite dar un título al fondo de la pantalla, arriba a la izquierda. En cuanto al parámetro `--title`, da un título al cuadro de diálogo.

Salga de la ayuda con la tecla Q. Teclee el comando siguiente completo:

```
$ dialog --backtitle "Elección de la fecha" --title "Calendario"  
--calendar "Seleccione la fecha" 0 0 01 06 2009
```

c - Haga un:

```
$ man -k dialog
```

Observe los resultados en la sección 1. Si Xdialog está instalado, su nombre debe aparecer. Acceda a la ayuda de Xdialog:

```
$ man Xdialog
```

Puede observar que este comando es casi idéntico a dialog. Si está bajo X, sustituya, en el punto anterior, dialog por Xdialog y valide:

```
$ Xdialog --backtitle "Elección de la fecha" --title "Calendario"  
--calendar "Seleccione la fecha" 0 0 01 06 2009
```

2. Para esta segunda parte, diríjase al sitio del proyecto de documentación de Linux, «The Linux Documentation Project», <http://tldp.org/>. Este sitio propone varias documentaciones: FAQ, HOWTO, guías (verdaderos libros), páginas del manual, y todo ello traducido o no.
  - a - Un HOWTO es un documento que describe cómo hacer tal o cual acción, paso a paso. Vea el HOWTO en español sobre el problema de Signal 11.

- b - En este HOWTO, encuentre la principal causa del origen de este mensaje.
- c - Como en este documento, sospecha de un fallo de una unidad de memoria. ¿Qué le aconseja este HOWTO?

*Respuestas:*

a - Haga clic a la izquierda, en Translated Howtos, luego en es en la lista del medio. Elija el formato deseado: Texto sin formato (una página grande en formato de texto), HTML sin formato, HTML por capítulos o PDF. Prefiera, de momento, el formato HTML en una página. Luego elija en la lista el sig11-HOWTO:

<http://www.ibiblio.org/pub/Linux/docs/HOWTO/translations/es/html-1page/Sig11.html>

b - Desde el principio, el HOWTO le indica que la principal causa de un bug que provoca un mensaje de tipo Signal 11 es un problema de hardware.

c - El HOWTO aconseja llevar a cabo varias pruebas, pero para la memoria habla de un test llamado memtest86. Como el HOWTO es antiguo, se dará cuenta de que es mejor utilizar memtest+.

# Requisitos y objetivos

## 1. Requisitos

- Haber configurado la BIOS para arrancar en el lector CD.
- Disponer de una fuente de instalación (CD, DVD, red).
- Saber manejar una consola o un terminal básico.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Instalar una distribución Debian.
- Instalar una distribución openSUSE.
- Gestionar el Red Hat Package Manager.
- Gestionar el gestor de paquetes YUM.
- Gestionar los paquetes Debian.
- Manejar los gestores APT y asociados.
- Instalar una aplicación desde los archivos fuente.
- Gestionar las librerías compartidas.

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

## **1. Preguntas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **Instalación**

- 1** ¿Desde qué sitio puede obtener la última versión de Debian?
- 2** ¿Cuál es la mejor interfaz para instalar Debian?
- 3** ¿Cuáles son las opciones de soporte de instalación que presenta el instalador de Debian?
- 4** ¿Por qué es importante que desde el principio de la instalación se elija bien el idioma?
- 5** ¿Por qué debe configurar la red (siempre que sea posible) durante la etapa de instalación, y no después?
- 6** Si un principiante instala Debian, ¿cuál sería el mejor método para particionar su disco duro?
- 7** Usted instala una distribución en su ordenador personal o en su puesto de trabajo en la empresa. ¿Cuál es la mejor elección para particionar los discos?
  - A - Una sola partición grande / ocupando todo el espacio.
  - B - Las particiones /, /usr, /var, /home, /boot.
  - C - Dos particiones: / y /home.
- 8** ¿Cuál es el sistema de ficheros elegido por defecto cuando usted crea sus particiones?
- 9** ¿Cuál es la partición que añade automáticamente el instalador?
- 10** ¿Para qué sirve la cuenta root?
- 11** ¿Por qué debe crear al menos una cuenta de usuario?
- 12** Durante la selección de los paquetes a instalar, se debe:
  - A - Instalarlo todo.
  - B - Instalar lo mínimo, a riesgo de tener que añadir después.
  - C - Instalar únicamente lo que necesita.
- 13** ¿Cuál es la misión de un gestor de arranque (bootloader)?
- 14** ¿Cuál es el interés de efectuar una prueba de memoria antes de una instalación?
  - A - Conocer la cantidad de memoria disponible.
  - B - Verificar la compatibilidad de sus pastillas de memoria con Linux.
  - C - Efectuar pruebas de velocidad.
  - D - Controlar la ausencia de errores físicos en las pastillas de memoria.
- 15** ¿Cómo se llama la herramienta de instalación de Fedora?
- 16** Si es usted usuario de Windows, ¿qué entorno gráfico le resultará más adecuado?
  - A - Gnome
  - B - KDE
  - C - XFCE

- D - TWM

## **Paquetes**

**17** ¿Qué es un paquete?

**18** Quiere instalar o suprimir un paquete y la resolución de dependencias se lo impide. Debe:

- A - Forzar la instalación o la desinstalación.
- B - Abandonar la instalación o la supresión.
- C - Resolver los problemas de dependencia antes de actuar.
- D - Encontrar otro paquete sin este problema.

## **RPM**

**19** ¿Qué significa RPM?

- A - Red Hat Package Manager.
- B - Real Player Media.
- C - Reverse Proxy Management.
- D - Ninguna de estas definiciones.

**20** Un fichero RPM se llama así: bash-3.2-112.x86\_64.rpm. ¿Qué afirmaciones son exactas?

- A - Es bash en versión 3.2.
- B - Este paquete funciona en las arquitecturas x86 y de 64 bits.
- C - Este paquete no se podrá instalar en una versión de 32 bits de Linux.
- D - Es la 112<sup>a</sup> versión de bash.

**21** Va a instalar todos los paquetes presentes en un directorio. Quiere conocer el progreso y los detalles de la instalación. ¿Qué comando debe teclear?

**22** Ha descargado un centenar de paquetes para realizar una actualización, pero no sabe cuáles son los que debe realmente actualizar. ¿Qué comando debe teclear?

**23** ¿Cómo desinstalar un paquete RPM?

- A - rpm -r
- B - rpm -e
- C - rpm -d
- D - rpmdel

**24** La sintaxis correcta para instalar un nuevo núcleo de Linux es:

- A - rpm -U
- B - rpm -F
- C - rpm -i
- D - Primero debe suprimir el antiguo.

**25** ¿Qué comando se debe teclear para saber a qué paquete pertenece el fichero /usr/bin/ksh?

**26** Ha descargado el paquete setserial-2.17-691.i586.rpm. Antes de instalarlo, quiere obtener el manifiesto de lo que contiene y la lista de sus ficheros. ¿Cómo conseguirlo?

**27** ¿Cómo saber si un paquete está instalado con sólo conocer una parte de su nombre?

- A - rpm -q "\*cadena\*"

- B - rpm -qa|grep cadena
- C - rpm -qial | grep cadena
- D - rpm -qf \*

**28** Tiene la sospecha de que se han modificado varios ficheros del paquete «file» ya instalados. ¿Qué comando permite saberlo?

## **DPKG**

**29** ¿Qué significa DPKG?

**30** ¿Cuál es el sufijo de un fichero de paquete Debian?

- A - deb
- B - dpkg
- C - rpm
- D - tgz

**31** Para instalar el paquete file-2.21-3.i386.deb, ¿qué comando debe usar?

**32** ¿Cómo listar todos los paquetes Debian instalados que contengan la palabra «passwd» en su nombre?

**33** ¿Cómo obtener la lista de todos los paquetes Debian instalados?

**34** ¿Cómo saber a qué paquete pertenece el fichero /bin/bash ?

**35** ¿Qué comando permite conocer el contenido de un paquete Debian instalado?

**36** ¿Qué herramienta permite convertir paquetes RPM en Debian y viceversa?

- A - rpmtodeb
- B - deb2rpm
- C - alien
- D - tar

## **APT**

**37** De las afirmaciones siguientes, ¿cuáles son verdaderas?

- A - APT gestiona las dependencias.
- B - Bajo Debian, APT gestiona también las RPM.
- C - APT permite precisar varios repositorios.
- D - APT no permite cambiar de versión de distribución.

**38** Dé la ruta del fichero que contiene la lista de los repositorios APT.

**39** ¿Cómo actualizar la base local de los repositorios?

**40** ¿Cómo actualizar de una vez todos los paquetes?

**41** ¿Cómo buscar un paquete preciso en los repositorios?

## **Fuentes**

**42** Si puede elegir, ¿es preferible utilizar un paquete o volver a compilar un programa?

**43** En general, ¿qué comando permite configurar las fuentes para preparar la compilación?

**44** Desea que una vez compilado el producto no se instale en /usr/local, sino en /opt. ¿Cómo conseguirlo?

**45** ¿Qué comando debe teclear para ejecutar efectivamente la compilación?

**46** Ha compilado su programa como usuario normal y sin errores. Ahora ejecuta un make install que le devuelve un error. ¿Qué está pasando?

**47** ¿Para qué sirve un fichero Makefile?

- A - Para compilar ficheros fuentes.
- B - Para ejecutar acciones, entre las cuales se halla la compilación.
- C - Para crear ficheros fuentes.
- D - Para lanzar el comando make.

### **Librerías compartidas**

**48** ¿Cuál es el verdadero nombre de las librerías compartidas bajo Linux?

- A - Shared Objects.
- B - Shared Libraries.
- C - Dynamic Objects.
- D - Dynamic Loadable Libraries.

**49** El programa «mygame» no se inicia, lo que indica que no encuentra una librería. ¿Cómo saber cuál es la librería que falta si se conoce la ruta de acceso al comando?

**50** Como usuario normal, ha compilado e instalado librerías en su directorio personal /home/user/lib. Sus programas no ven sus librerías. ¿Qué tiene que hacer?

**51** Como administrador, debe añadir el directorio /opt/lib a la lista de las rutas de librerías para todo el sistema. ¿Cómo proceder en dos etapas?

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /51

Para este capítulo, su resultado mínimo debe ser de 38 respuestas acertadas (38/51).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- Instalación.
- Paquetes.
- RPM.
- DPKG.
- APT.
- Fuentes.
- Librerías compartidas.

## **3. Respuestas**

### **Instalación**

**1** ¿Desde qué sitio puede obtener la última versión de Debian?

<http://www.debian.org>

**2** ¿Cuál es la mejor interfaz para instalar Debian?

*La interfaz en modo texto.*

**3** ¿Cuáles son las opciones de soporte de instalación que presenta el instalador de Debian?

*Las teclas de función de [F1] a [F10] proporcionan la ayuda y todos los parámetros que se pueden pasar al boot. La tecla [Intro] ejecuta la instalación.*

**4** ¿Por qué es importante que desde el principio de la instalación se elija bien el idioma?

*Su elección influye en el contenido de las variables «LOCAL» después de la instalación: el idioma de los comandos, el manual, los números, etc.*

**5** ¿Por qué debe configurar la red (siempre que sea posible) durante la etapa de instalación, y no después?

*Para poder acceder directamente a las actualizaciones y los repositorios de programas durante la instalación.*

**6** Si un principiante instala Debian, ¿cuál sería el mejor método para particionar su disco duro?

*El método asistido en modo automático.*

**7** Usted instala una distribución en su ordenador personal o su puesto de trabajo en empresa. ¿Cuál es la mejor elección para particionar los discos?

- A - Una sola partición grande / ocupando todo el espacio.
- B - Las particiones /, /usr, /var, /home, /boot.
- C - Dos particiones: / y /home.

*La opción C. En caso de reinstalación del sistema, sólo sobreescribirá / y se conservarán sus datos.*

**8** ¿Cuál es el sistema de ficheros elegido por defecto cuando usted crea sus particiones?

*El sistema de ficheros ext3 o ext4.*

**9** ¿Cuál es la partición que añade automáticamente el instalador?

*La partición de swap, calculada según la memoria de la cual dispone.*

**10** ¿Para qué sirve la cuenta root?

*Es la cuenta que permite administrar su sistema Linux.*

**11** ¿Por qué debe crear al menos una cuenta de usuario?

*Porque la cuenta de administrador no debe destinarse a un uso corriente del sistema.*

**12** Durante la selección de los paquetes a instalar, se debe:

- A - Instalarlo todo.
- B - Instalar lo mínimo, a riesgo de tener que añadir después.
- C - Instalar únicamente lo que necesita.

*B o C: En todos los casos, podrá añadir más adelante lo que le falta. Pero lo mejor sigue siendo partir desde la C, a riesgo de tener que añadir o quitar algunos paquetes más adelante. No lo instale todo, esto vuelve inútilmente pesada su configuración.*

**13** ¿Cuál es la misión de un gestor de arranque (bootloader)?

*Sirve para seleccionar y ejecutar uno de los sistemas operativos presentes.*

**14** ¿Cuál es el interés de efectuar una prueba de memoria antes de una instalación?

- A - Conocer la cantidad de memoria disponible.
- B - Verificar la compatibilidad de sus pastillas de memoria con Linux.
- C - Efectuar pruebas de velocidad.
- D - Controlar la ausencia de errores físicos en las pastillas de memoria.

*D. La memoria es el componente más sensible y la causa de muchos bugs en caso de que presente algún defecto. Debería probarla antes de instalar Linux.*

**15** ¿Cómo se llama la herramienta de instalación de Fedora?

*Anaconda.*

**16** Si es usted usuario de Windows, ¿qué entorno gráfico le resultará más adecuado?

- A - Gnome
- B - KDE
- C - XFCE
- D - TWM

*B, KDE como primera elección. Si no, A, Gnome.*

## **Paquetes**

**17** ¿Qué es un paquete?

*Es un fichero que contiene un producto a instalar, las reglas de dependencias y los tratos pre y postinstalación que se aplican a él.*

**18** Quiere instalar o suprimir un paquete y la resolución de dependencias se lo impide. Debe:

- A - Forzar la instalación o la desinstalación.
- B - Abandonar la instalación o la supresión.
- C - Resolver los problemas de dependencia antes de actuar.
- D - Encontrar otro paquete sin este problema.

*C. Evite forzar una instalación o una supresión, ya que otros productos podrían dejar de funcionar.*

## **RPM**

**19** ¿Qué significa RPM?

- A - Red Hat Package Manager.
- B - Real Player Media.
- C - Reverse Proxy Management.
- D - Ninguna de estas definiciones.

A.

**20** Un fichero RPM se llama así: bash-3.2-112.x86\_64.rpm. ¿Qué afirmaciones son exactas?

- A - Es bash en versión 3.2.
- B - Este paquete funciona en las arquitecturas x86 y de 64 bits.
- C - Este paquete no se podrá instalar en una versión de 32 bits de Linux.
- D - Es la 112<sup>a</sup> versión de bash.

A y C. Se trata de bash en versión 3.2, compilada e instalable en x86\_64 (arquitectura Intel o AMD de 64 bits). Es la 112<sup>a</sup> versión de este paquete, no de bash.

**21** Va a instalar todos los paquetes presentes en un directorio. Quiere conocer el progreso y los detalles de la instalación. ¿Qué comando debe teclear?

*rpm -ivh \*.rpm : la i instala, la v indica el nombre del paquete, la h sitúa un indicador.*

**22** Ha descargado un centenar de paquetes para realizar una actualización, pero no sabe cuáles son los que debe realmente actualizar. ¿Qué comando debe teclear?

*rpm -F \*.rpm : la -F, de Freshen, sólo actualiza los paquetes ya presentes.*

**23** ¿Cómo desinstalar un paquete RPM?

- A - rpm -r
- B - rpm -e
- C - rpm -d
- D - rpmdel

B. La -e desinstala un paquete.

**24** La sintaxis correcta para instalar un nuevo núcleo de Linux es:

- A - rpm -U
- B - rpm -F
- C - rpm -i
- D - Primero debe suprimir el antiguo.

C. Este comando instala el nuevo respetando el antiguo. En caso de problema, podrá volver a ejecutar el antiguo núcleo.

**25** ¿Qué comando se debe teclear para saber a qué paquete pertenece el fichero /usr/bin/ksh?

*rpm -qf /usr/bin/ksh.*

**26** Ha descargado el paquete setserial-2.17-691.i586.rpm. Antes de instalarlo, quiere obtener el manifiesto de lo que contiene y la lista de sus ficheros. ¿Cómo conseguirlo?

*rpm -qilp setserial-2.17-691.i586.rpm. La i facilita la descripción, la l da la lista de los ficheros, la p precisa un fichero de paquete.*

**27** ¿Cómo saber si un paquete está instalado con sólo conocer una parte de su nombre?

- A - rpm -q "\*cadena\*"
- B - rpm -qalgrep cadena

- C - rpm -qial | grep cadena
- D - rpm -qf \*

*B: liste todos los rpms y haga una selección en parte del nombre.*

**28** Tiene la sospecha de que se han modificado varios ficheros del paquete «file» ya instalados. ¿Qué comando permite saberlo?

*rpm -V file*

### **DPKG**

**29** ¿Qué significa DPKG?

*Debian Package.*

**30** ¿Cuál es el sufijo de un fichero de paquete Debian?

- A - deb
- B - dpkg
- C - rpm
- D - tgz

*A.*

**31** Para instalar el paquete file-2.21-3.i386.deb, ¿qué comando debe usar?

*dpkg -i file-2.21-3.i386.deb.*

**32** ¿Cómo listar todos los paquetes Debian instalados que contengan la palabra «passwd» en su nombre?

*dpkg -l "\*nombre\*" | grep ^ii.*

**33** ¿Cómo obtener la lista de todos los paquetes Debian instalados?

*dpkg --get-selections.*

**34** ¿Cómo saber a qué paquete pertenece el fichero /bin/bash?

*dpkg -S /bin/bash.*

**35** ¿Qué comando permite conocer el contenido de un paquete Debian instalado?

*dpkg -L package.*

**36** ¿Qué herramienta permite convertir paquetes RPM en Debian y viceversa?

- A - rpmtodeb
- B - deb2rpm
- C - alien
- D - tar

*C, alien.*

### **APT**

**37** De las afirmaciones siguientes, ¿cuáles son verdaderas?

- A - APT gestiona las dependencias.
- B - Bajo Debian, APT gestiona también las RPM.
- C - APT permite precisar varios repositorios.
- D - APT no permite cambiar de versión de distribución.

*A y C. APT no gestiona las rpm por defecto bajo Debian, y permite actualizar Debian hacia otra versión.*

**38** Dé la ruta del fichero que contiene la lista de los repositorios APT.

*/etc/apt/sources.list.*

**39** ¿Cómo actualizar la base local de los repositorios?

*apt-get update.*

**40** ¿Cómo actualizar de una vez todos los paquetes?

*apt-get upgrade.*

**41** ¿Cómo buscar un paquete preciso en los repositorios?

*apt-cache search.*

## **Fuentes**

**42** Si puede elegir, ¿es preferible utilizar un paquete o volver a compilar un programa?

*Debería instalar el paquete si existe, excepto si la recompilación le aporta nuevas funcionalidades.*

**43** En general, ¿qué comando permite configurar las fuentes para preparar la compilación?

*Es el comando configure.*

**44** Desea que una vez compilado el producto no se instale en /usr/local, sino en /opt. ¿Cómo conseguirlo?

*Modifique el prefijo de instalación con --prefix: configure --prefix=/opt.*

**45** ¿Qué comando debe teclear para ejecutar efectivamente la compilación?

*El comando make.*

**46** Ha compilado su programa como usuario normal y sin errores. Ahora ejecuta un make install que le devuelve un error. ¿Qué está pasando?

*Está intentando instalar el programa en un lugar prohibido. Abra una sesión root mediante el comando su o sudo antes de ejecutar el comando.*

**47** ¿Para qué sirve un fichero Makefile?

- A - Para compilar ficheros fuentes.
- B - Para ejecutar acciones, entre las cuales se halla la compilación.
- C - Para crear ficheros fuentes.
- D - Para lanzar el comando make.

*B. El comando make no compila nada por sí mismo. Ejecuta el compilador (u otro comando) según las reglas que usted especifica en el fichero Makefile.*

## **Librerías compartidas**

**48** ¿Cuál es el verdadero nombre de las librerías compartidas bajo Linux?

- A - Shared Objects.
- B - Shared Libraries.
- C - Dynamic Objects.
- D - Dynamic Loadable Libraries.

*A: Shared Objects, objeto compartido.*

**49** El programa «mygame» no se inicia, lo que indica que no encuentra una librería. ¿Cómo saber cuál es la librería que falta si se conoce la ruta de acceso al comando?

*Teclee ldd /ruta/mygame*

**50** Como usuario normal, ha compilado e instalado librerías en su directorio personal /home/user/lib. Sus programas no ven sus librerías. ¿Qué tiene que hacer?

*Tiene que precisar mediante la variable LD\_LIBRARY\_PATH la ruta de acceso a sus librerías.*

**51** Como administrador, debe añadir el directorio /opt/lib a la lista de las rutas de librerías para todo el sistema. ¿Cómo proceder en dos etapas?

*Tiene que añadir primero /opt/lib en el fichero /etc/ld.so.conf y luego ejecutar el comando ldconfig.*

# Prácticas

## 1. Esquema de particionamiento

Esta práctica tiene como objetivo determinar el mejor esquema de particionamiento posible, que es lo que presenta la mayor dificultad para un principiante. La práctica está pensada para todas las distribuciones. Usted dispone en su PC de un disco de 160 GB, de los cuales 40 están ocupados por otro sistema. Su máquina dispone de 2 GB de memoria RAM. Le quedan 120 GB de espacio en disco. ¿Cómo puede repartirlos sabiendo que quiere separar sus datos y el sistema?

1. ¿Qué tamaño debe tener la partición de intercambio SWAP?  
Se utiliza la partición de intercambio cuando Linux no dispone de bastante espacio en memoria RAM para trabajar con todos sus datos. Se mueven de forma temporal los datos menos urgentes a la memoria virtual (esta partición de intercambio) para liberar memoria para otros datos.  
Al disponer su sistema de 2 GB de memoria, sólo puede prever 2 GB de swap. Quedan sólo 118 GB de espacio en disco para el resto.
2. ¿Que espacio reservar a la raíz de sistema /?  
Incluso si tuviera que instalar todos los productos presentes en el DVD de instalación, el total no alcanzaría 10 GB. Pero dos cosas debe tener en cuenta: siempre puede añadir software procedente de otras fuentes (nuevos repositorios, instalación manual, etc.) y la actividad sobre el sistema puede tener como consecuencia que los directorios /var y /tmp crezcan muchísimo. Si dispone de bastante espacio, ¿por qué no reservar 20 o 30 GB a la raíz?  
Cojamos por ejemplo 20 GB.  
Quedan 98 GB.
3. ¿Qué espacio reservar al /home?  
La partición que contiene /home es la que contiene sus datos o la de otros usuarios: la que ocupa más espacio, entre las fotos, la música, las películas, los documentos de trabajo, etc.  
Reserve los 98 GB restantes. El disco está totalmente particionado.
4. ¿Es útil crear una partición extendida?  
Tiene que crear tres particiones, además de la que ya existe. O sea, un total de cuatro particiones. cuatro es el máximo de particiones primarias. Pero piense que puede necesitar reducir, suprimir o volver a crear particiones. En este caso, ya se habría alcanzado el límite.  
Sea prudente y establezca una partición extendida donde ubicar particiones lógicas.
5. ¿Cuál es el esquema final del disco?  
Partición primaria 1: el OS ya presente, 40 GB.  
Partición extendida:
  - Partición lógica 1: /, 20 GB.
  - Partición lógica 2: /home, 98 GB.
  - Partición lógica 3: swap, 2 GB.Los 160 GB están todos particionados.

## 2. Gestión de los RPM

El objetivo de esta práctica es trabajar con la base de datos de paquetes RPM ya instalados en su equipo e instalar otros nuevos. El equipo o la máquina virtual debe disponer de una distribución basada en RPM (Red Hat, Fedora, Mandriva, openSUSE, etc.).

1. Determine el número de paquetes RPM actualmente instalados en su puesto de trabajo.  
*Respuesta:*  
Ejecute el comando siguiente:

```
$ rpm -qa | wc -l
```

En el puesto del autor, devuelve: 1684

2. Verifique que el paquete coreutils está presente en su sistema, luego determine de qué sirve con la ayuda de su manifiesto. ¿Puede conseguir obtener sólo la descripción y nada más? Lea la página del manual correspondiente para ampliar sus conocimientos. Primero, consulte la base RPM sobre este paquete para obtener la información:

```
$ rpm -qi coreutils
```

En caso de error, seguramente no se instalará el paquete. Si no, lea el contenido del campo Descripción.

Segundo, lea la sección del manual de rpm dedicada al formato de salida. El parámetro -q acepta un formato de salida que usted puede formatear con --queryformat. El formato se especifica así: %{CAMPO} con el campo en mayúsculas:

```
$ rpm -q --queryformat=%{DESCRIPTION}
```

3. Intente suprimir el paquete coreutils. ¿Puede proporcionar la lista de las dependencias que le impiden hacerlo?

Intente como root:

```
# rpm -e coreutils
```

Ha obtenido la lista de todos los paquetes que impiden su desinstalación: ¡varios centenares! Observe la existencia del parámetro -R, que muestra de qué depende el propio paquete, y el -provides, que facilita el nombre de los elementos proporcionados por el paquete.

```
$ rpm -q --provides coreutils
fileutils
sh-utils
stat
textutils
coreutils = 6.9-43
```

4. Descargue el paquete RPM de tuxpaint que encontrará en <http://www.tuxpaint.org/download/linux-rpm/>, elija la última versión. Aunque está indicada para Fedora, funciona también en Mandriva u openSUSE. Instale este paquete indicando que quiere ver su nombre y una barra de progreso.

Instale el paquete con los parámetros -i, -v y -h:

```
# rpm -ivh tuxpaint.xxxxxxx.rpm
```

5. Si hubiera estado instalado el paquete, ¿cómo habría podido actualizarlo? Sabiendo que ya está instalado ahora, intente actualizar este paquete sin condiciones. ¿En qué caso puede ser necesario? Finalmente, suprímalo.

Puede actualizar el paquete con los parámetros -U o -F. Observe que hubiera podido instalar el paquete directamente con -U:

```
# rpm -Uvh tuxpaint.xxxxxxx.rpm
```

Pero si ya está instalada la misma versión de este paquete, eso no funcionará. Podría necesitar hacerlo si se han suprimido algunos de los ficheros de este paquete: su supresión, incluso completa, no suprime el rpm de la base local. Indique la opción --force.

```
# rpm -Uvh --force tuxpaint.xxxxxxx.rpm
```

### 3. Gestión de DPKG y APT

El objetivo de esta práctica es trabajar con la base de datos de los paquetes dpkg instalados en su máquina, instalar otros nuevos y utilizar APT. Su equipo o máquina virtual debe disponer de una distribución de tipo Debian o Ubuntu.

1. Conteste a las preguntas 1 a 4 de la práctica anterior, pero con los comandos y paquetes DPKG equivalentes: coreutils está presente con el mismo nombre y tuxpaint dispone de paquetes Debian en <http://www.tuxpaint.org/download/linux-debian/> (es posible que deba instalar los paquetes adicionales proporcionados).  
a - Hay que filtrar la lista de los paquetes instalados. Por defecto, dpkg facilita la lista de todos los paquetes conocidos, entre los cuales se hallan, los instalados, que empiezan por «ii»:

```
$ dpkg -l | grep ^ii | wc -l
```

En la máquina de prueba del autor, hay 695 paquetes instalados.

b - La opción -l de dpkg acepta un filtro como parámetro:

```
$ dpkg -l "*coreutils"
```

Es posible que encuentre dos paquetes con este nombre; por lo tanto, deberá leer los resultados o buscar una correspondencia exacta:

```
$ dpkg -l coreutils
```

Para obtener los detalles del paquete que tiene ya instalado, hay que dirigirse al manual, donde se le informa de que es posible utilizar el comando **dpkg-query** y el parámetro **-W**:

```
$ dpkg-query -W coreutils
```

Pero falta la descripción. El manual de dpkg-query proporciona una información adicional: puede modificar el formato de salida con **-f**:

```
$ dpkg-query -W -f='${Description}' coreutils
```

c - Para suprimir un paquete Debian, utilice la opción **-r**:

```
# dpkg -r coreutils
```

Obtendrá errores:

```
dpkg: error de tratamiento de coreutils (--remove):  
  Es un paquete indispensable - no se debe suprimir.  
Se encontró errores durante la ejecución:  
coreutils
```

Para conocer qué paquetes dependen de coreutils, utilice de nuevo dpkg-query:

```
# dpkg-query -W -f='${Provides}' coreutils  
textutils, shellutils, fileutils
```

d - Para instalar un paquete Debian, utilice el parámetro **-i**:

```
# dpkg -i tuxpaint.xxxxxxx.dpkg
```

Como ya se ha indicado en este capítulo, no hay un método directo equivalente a rpm para la actualización de un paquete. Si ya está instalado el paquete, **-i** lo actualizará. Le corresponde a usted comprobar antes si efectivamente está ya instalado (para ello, vea la respuesta a la primera pregunta).

2. APT es un gestor de metapaquetes: gestiona las dependencias en lugar de usted y trabaja con repositorios, y no con paquetes individuales. Tuxpaint está presente en los repositorios de

los backports. El objetivo es gestionar este repositorio y las instalaciones asociadas.

a - Añada la línea «deb http://www.backports.org/debian etch-backports main contrib non-free» en el fichero de los repositorios. ¿Cuál es la URL del repositorio contrib para una arquitectura i386?

b - Actualice la base de datos local de APT.

c - Instale únicamente tuxpaint.

d - Actualice la totalidad de su distribución con los posibles nuevos paquetes disponibles dentro de sus repositorios.

*Respuesta:*

a - Debe modificar el fichero /etc/apt/sources.list y añadir la línea «deb http://www.backports.org/debian etch-backports main contrib non-free». La URL completa del repositorio contrib para los i386 es http://www.backports.org/debian/dists/etch-backports/contrib/binary-i386/.

b - Actualice la base local APT con el comando siguiente:

```
# apt-get update
```

c - Instale únicamente tuxpaint de la manera siguiente:

```
# apt-get install tuxpaint
```

Observe que, a diferencia de lo visto en la primera pregunta, APT gestiona las dependencias y va a instalar tuxpaint, así como las dependencias asociadas.

d - Actualice su sistema con:

```
# apt-get upgrade
```

## 4. Las fuentes

Esta práctica propone compilar las fuentes de un gestor de ficheros KDE llamado Beesoft Commander, disponible en <http://www.kde-apps.org/content/show.php/Beesoft+Commander?content=374351>

1. Descargue las fuentes en el formato tar.gz en su directorio personal, descomprima la carpeta resultante. Cambie el nombre del directorio de las fuentes a bsc\_sources y entre en el directorio de las fuentes.

El vínculo de descarga apunta al sitio beesoft. Coloque el fichero de su directorio personal. Para descomprimir las fuentes, teclee:

```
$ tar xvzf bsc_4.0.07.beta_src.tar.gz
```

Cambie el nombre del directorio resultante de la descompresión:

```
$ mv bsc bsc_sources
```

Acceda al directorio:

```
$ cd bsc_source
```

2. La compilación de este producto se ejecuta con un script llamado ./install.pl. Configure las fuentes para una instalación en /usr/local y ejecute la compilación. Edite el fichero install.pl. Sustituya inmediatamente las rutas /usr por /usr/local. Guarde. Luego ejecute el comando:

```
$ ./install.pl
```

Después de la compilación, install.pl intenta instalar el programa en la ruta indicada. Pero no funciona, ya que sólo root puede hacerlo. Por lo tanto, va a instalar el programa en \$HOME/bsc, para su usuario. Para instalarlo para todos, hubiera sido necesario compilar bsc con los derechos root, lo que no es normal.

3. Si bsc se ha compilado correctamente, no debe haber problemas para ejecutarlo. En otro caso, ¿qué puede haber pasado?  
BSC depende de las librerías de desarrollo QT4. ¿Están instaladas correctamente en su máquina? Si no, utilice su gestor de paquete para hacerlo, y vuelva a ejecutar la compilación.

## 5. Librerías compartidas

En esta práctica, va a gestionar las librerías compartidas vinculadas al programa compilado en la práctica anterior.

1. Compruebe a qué librerías compartidas está vinculado el programa bsc. ¿Encuentra una librería llamada libQT4Gui.so.4?

El comando **ldd** permite ver todas las librerías compartidas utilizadas por un programa:

```
$ ldd bsc
```

Aíslle la línea que busca:

```
$ ldd bsc | grep -i qtgui
```

Debe devolver una única línea. En la máquina de prueba la librería está ubicada en la ruta /usr/lib.

2. Partiendo del principio de que ningún programa utiliza la librería compartida libQT4Gui.so.4, diríjase a su ubicación como root y mueva todos los ficheros asociados en /tmp/lib. Ejecute luego el comando **ldconfig**. ¿Puede ejecutar bsc? Explíquelo.

Cree el directorio /tmp/lib:

```
# mkdir /tmp/lib
```

Diríjase a /usr/lib (o la ruta de la librería):

```
# cd /usr/lib
```

Mueva los ficheros:

```
# mv libQtGui.so* /tmp/lib
```

Ejecute bsc. No se inicia:

```
# ./bsc
./bsc: error while loading shared libraries: libQtGui.so.4: cannot
open shared object file: No such file or directory
```

El comando **ldd** le informa de la desaparición de la librería movida:

```
libQtGui.so.4 => not found
```

3. Edite el fichero de configuración /etc/ld.so.conf y añádale la ruta /tmp/lib. Ejecute de nuevo Idconfig y luego vuelva a iniciar bsc. ¿Qué está pasando?  
Al añadir la ruta /tmp/lib y luego actualizar la caché del cargador dinámico con ldconfig, la librería vuelve a estar accesible.

```
# vi /etc/ld.so.conf
# ldconfig
# ldd ./bsc
libQtGui.so.4 => /tmp/lib/libQtGui.so.4 (0xb77bc000)
```

Añada /tmp/lib en una nueva línea y guarde el fichero.  
El programa funciona de nuevo.

4. Vuelva a ponerlo todo en su estado inicial.  
Mueva la librería hacia su posición de origen, suprima /tmp/lib de ld.so.conf y vuelva a ejecutar ldconfig. Suprima /tmp/lib.

# Instalar una Debian

## 1. Soporte de instalación

A continuación se explica, paso a paso, cómo instalar una Debian. Se trata de la última versión estable, o sea, la 6.0.2 Squeeze, en vigor durante la redacción de este libro. La instalación la haremos en modo texto porque representa el mejor método para Debian (existe un instalador gráfico poco utilizado). Si desea efectuar una instalación remota, debe obtener la imagen ISO (en el libro usamos la versión de 64 bits) correspondiente mediante la red (Internet), llamada netinst y disponible en el sitio de Debian:

<http://cdimage.debian.org/debian-cd/6.0.2.1/amd64/iso-cd/debian-6.0.2.1-amd64-netinst.iso>

Guarde esta imagen en un CD-ROM o en una llave USB. Para las necesidades de este libro, se instaló Debian en una máquina virtual VirtualBox 4, que es un producto gratuito y del que existe una versión libre.

## 2. Boot en el soporte



*Boot del soporte de Debian*

Configure su ordenador para que se inicie en modo instalación. En el momento de ejecutarse, usted tendrá acceso a una línea de comandos que permite dar paso a la instalación presionando [Intro] o las teclas de [F1] a [F10] para acceder a pantallas de ayuda adicionales. Considere las opciones en línea de comandos según su máquina (de [F5] a [F10]), ya que en algunos casos puntuales puede ser necesario modificar los valores que se pasan al núcleo de Linux para un buen funcionamiento.

Welcome to Debian GNU/Linux!

This is a Debian 6.0 (squeeze) installation CD-ROM.  
It was built 20110628-12:58; d-i 20110106+squeeze3.

**HELP INDEX**

**KEY      TOPIC**

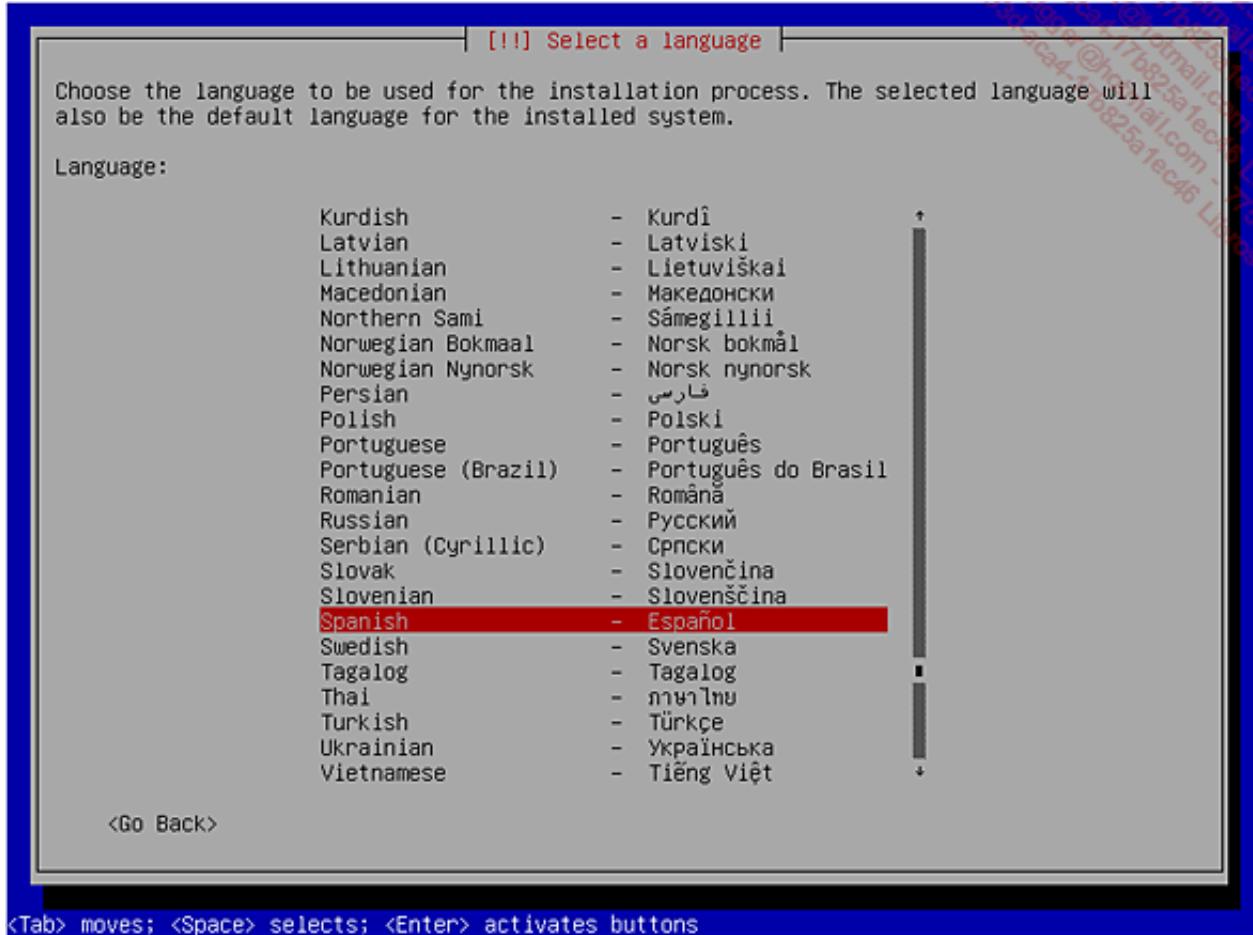
- <**F1**> This page, the help index.
- <**F2**> Prerequisites for installing Debian.
- <**F3**> Boot methods for special ways of using this CD-ROM
- <**F4**> Additional boot methods; rescue mode.
- <**F5**> Special boot parameters, overview.
- <**F6**> Special boot parameters for special machines.
- <**F7**> Special boot parameters for selected disk controllers.
- <**F8**> Special boot parameters for the install system.
- <**F9**> How to get help.
- <**F10**> Copyrights and warranties.

Press F2 through F10 for details, or ENTER to boot: \_

*Opciones especiales del boot*

Presione la tecla [Intro] para iniciar la instalación. Comprobará que el propio instalador funciona bajo Linux y que le muestra todas las líneas de inicio del núcleo de Linux. Finalmente, se visualiza la primera pantalla del instalador.

### 3. Elección de los idiomas y países

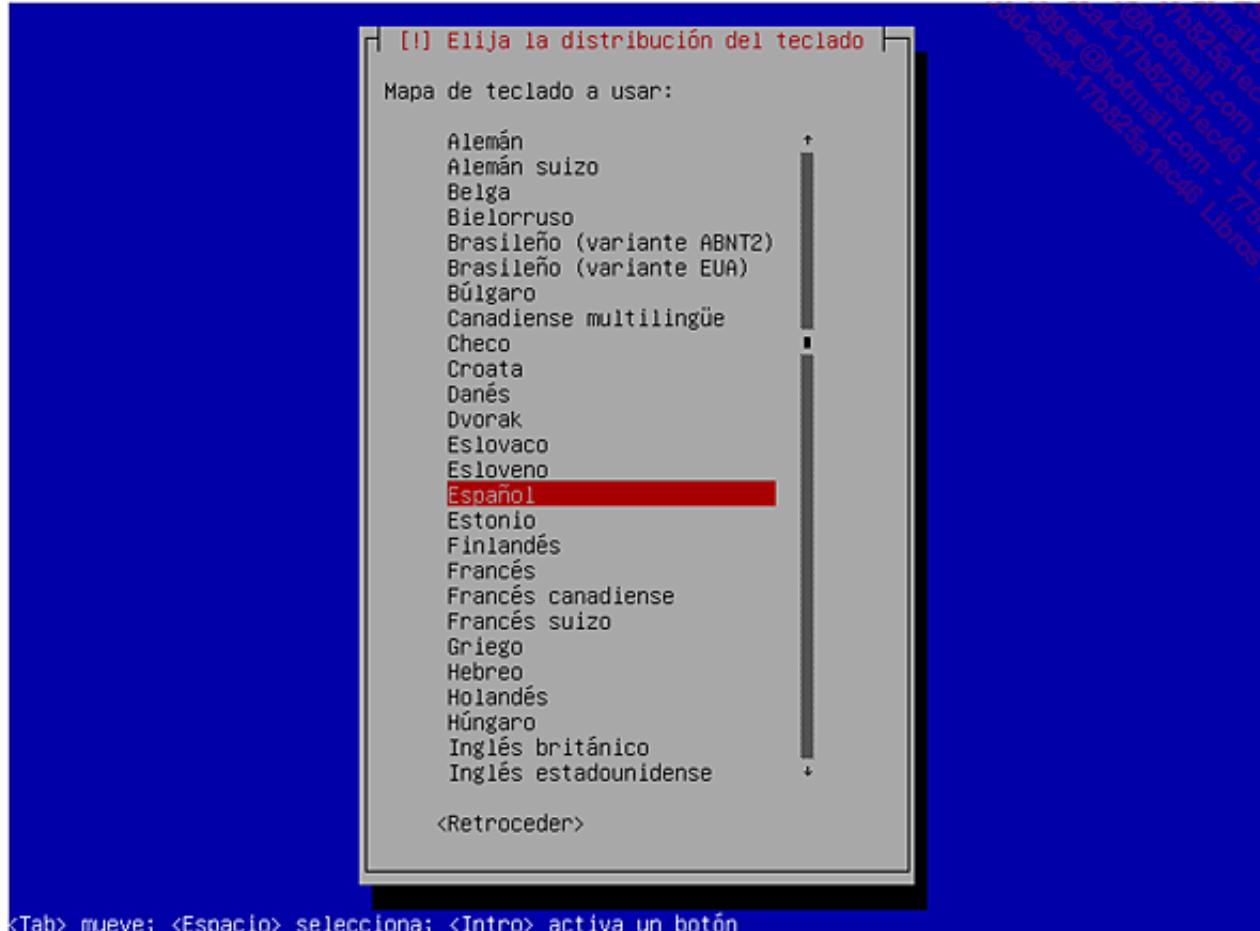


<Tab> moves; <Space> selects; <Enter> activates buttons

### *Eleción del idioma predeterminado*

Las siguientes tres pantallas le permiten elegir:

- El idioma utilizado por el proceso de instalación. Navegue con las flechas, pulse [Intro] para seguir. A continuación, se utiliza el español.
- Según el idioma inicial elegido, Debian le propone un país. Según el país, Debian configura los formatos de fecha, hora, alfabeto, formatos numéricos y monetarios, etc.
- Para terminar, elija su tipo de teclado. Para España, es el es-latin9.

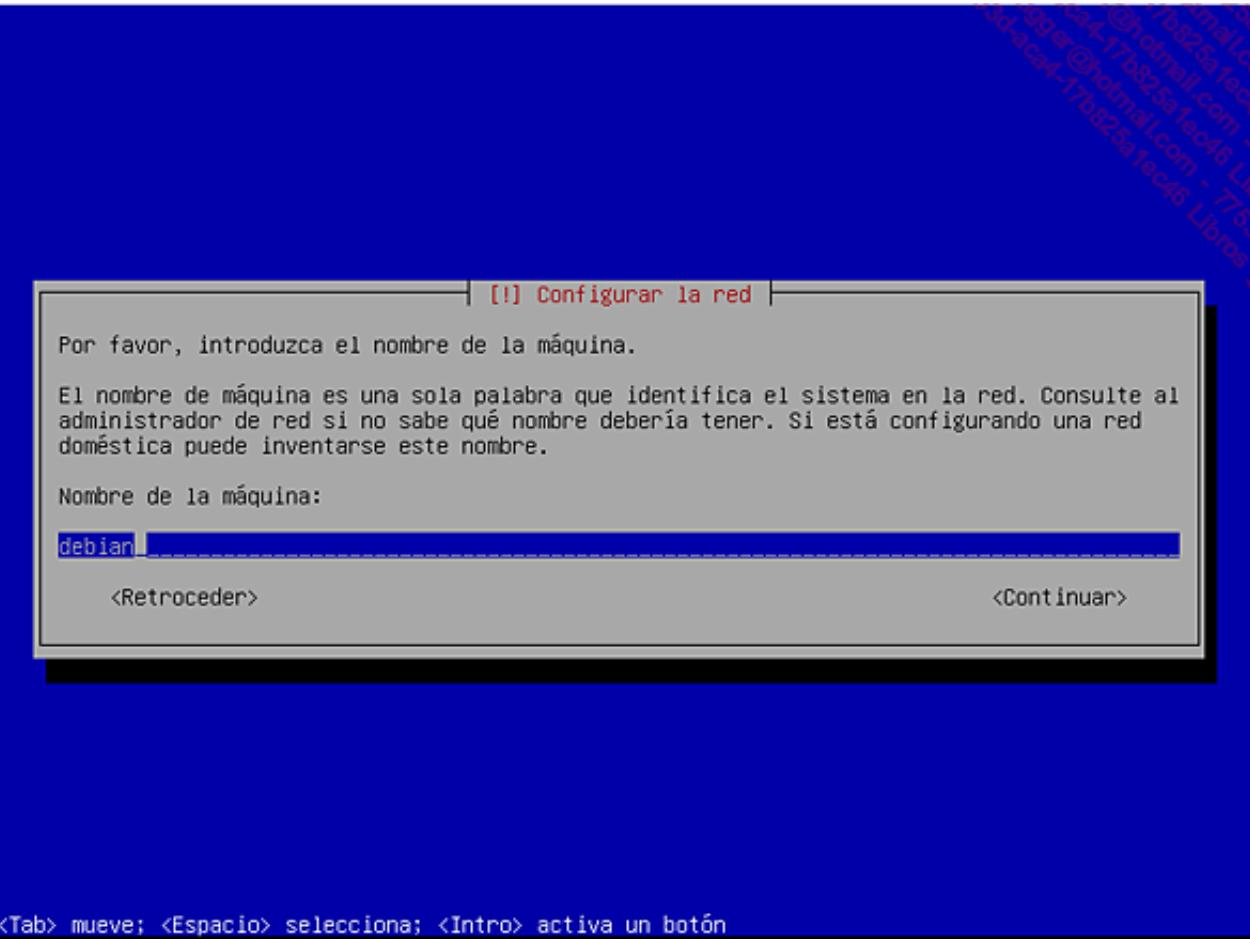


<Tab> mueve; <Espacio> selecciona; <Intro> activa un botón

*Elección del tipo de teclado*

## 4. Anfitriones y usuarios

Las etapas siguientes consisten en introducir un nombre de anfitrión (el nombre de la máquina en la red) y el nombre del dominio. Si su equipo no pertenece a ningún dominio, deje el campo vacío.

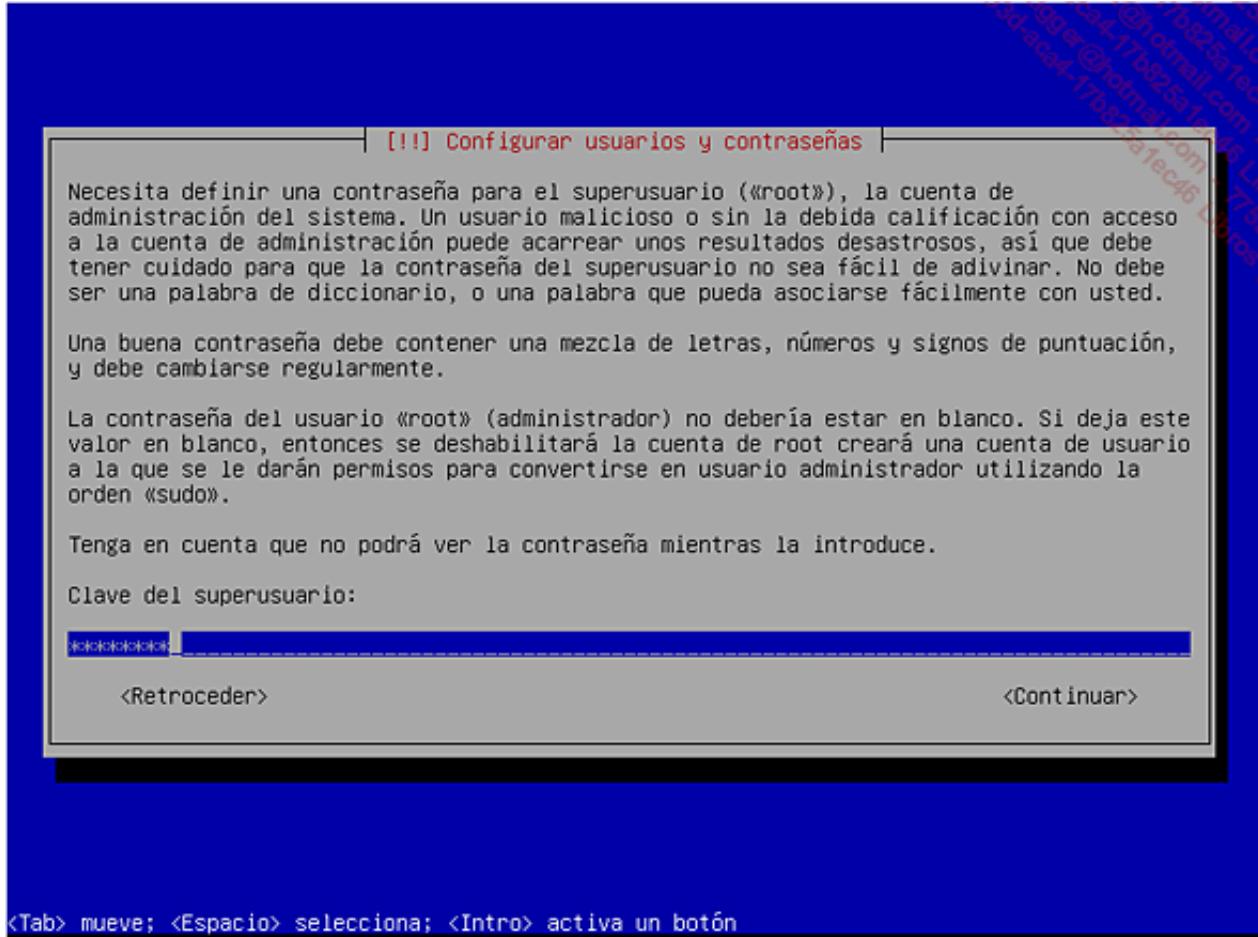


*Nombre de anfitrión de la máquina*

## 5. Cuentas root y usuarios

Ahora debe introducir la contraseña del administrador root de la máquina. Se la pedirá dos veces, por confirmación. ¡No la pierda! Aunque existan algunos métodos para reinicializar la contraseña, o «craquearla» mediante una herramienta especializada, no hay ninguna manera de recuperar la contraseña de origen. Debian incluso le avisa de que puede rechazar una contraseña si es demasiado simple.

Luego debe crear al menos un usuario «normal». Deberá introducir el nombre completo de la persona y Debian le propondrá un login. Introduzca luego la contraseña. Puede crear varios usuarios ahora, pero nada le impide hacerlo después de la instalación.



<Tab> mueve; <Espacio> selecciona; <Intro> activa un botón

*Introducción de la contraseña de root*

## 6. Partitionar los discos

Con el fin de simplificar, diremos que dispone de tres métodos principales para particionar sus discos:

- Un método asistido (o automático) al utilizar el particionamiento clásico (véase para ello el capítulo Los discos y el sistema de ficheros).
- Un método asistido que propone el LVM (*Logical Volume Manager*).
- Un método manual.

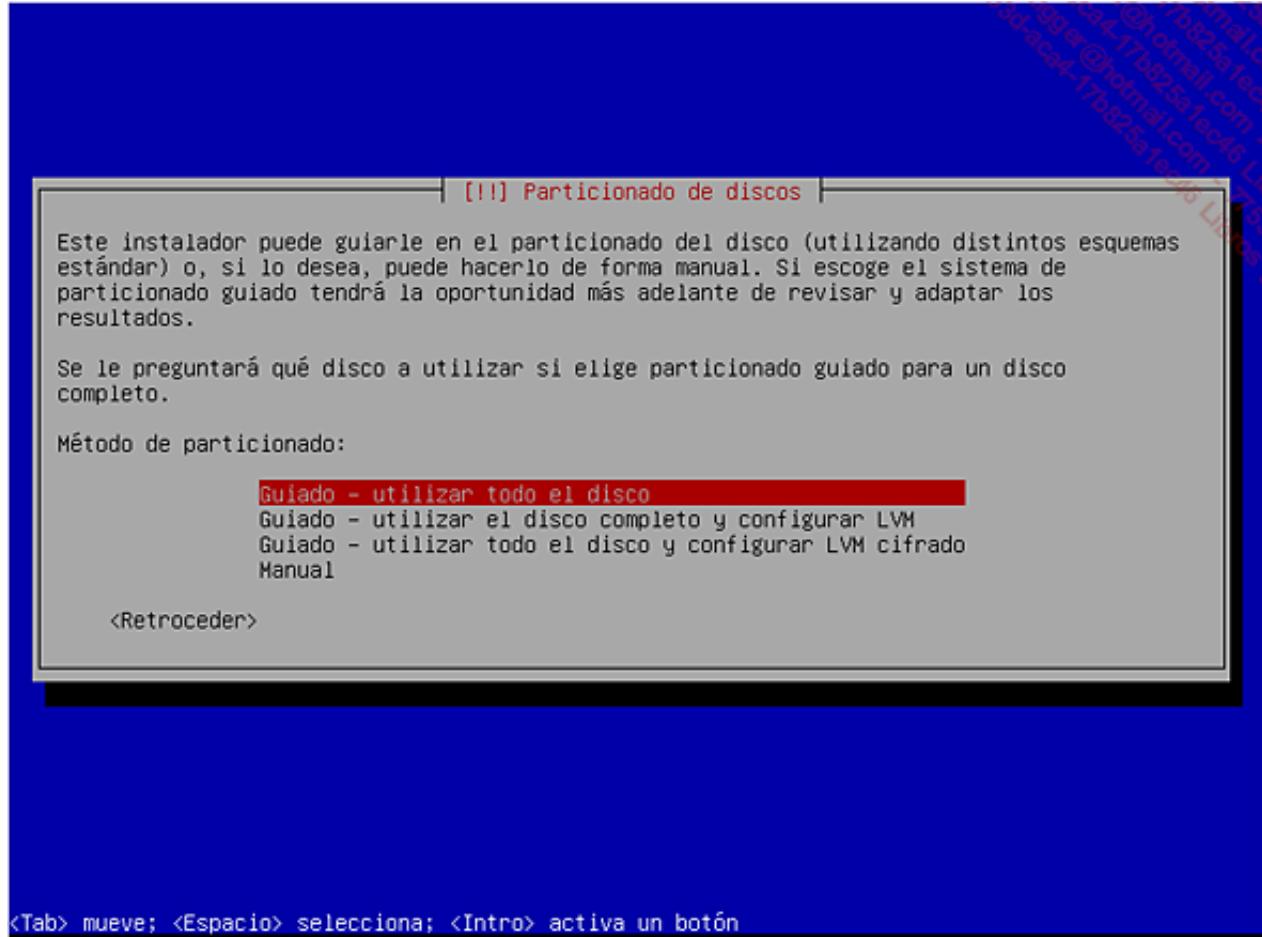
El método asistido clásico, en el caso de una nueva instalación, da buenos resultados. Si vuelve a instalar una máquina, o instala Debian en una máquina que dispone ya de particiones que contienen las carpetas personales, por ejemplo, pase a un particionado personalizado.

El LVM funciona agrupando discos físicos o particiones (llamados volúmenes físicos) en un solo espacio grande (llamado grupo de volúmenes), en el cual usted puede seccionar espacios lógicos a voluntad (llamados volúmenes lógicos), ampliarlos, reducirlos, etc.

---

 Sin embargo, debería considerar la solución LVM en el contexto de un servidor de empresa o si piensa agregar luego discos en su máquina para añadir espacio de almacenamiento. El LVM aporta mucha flexibilidad.

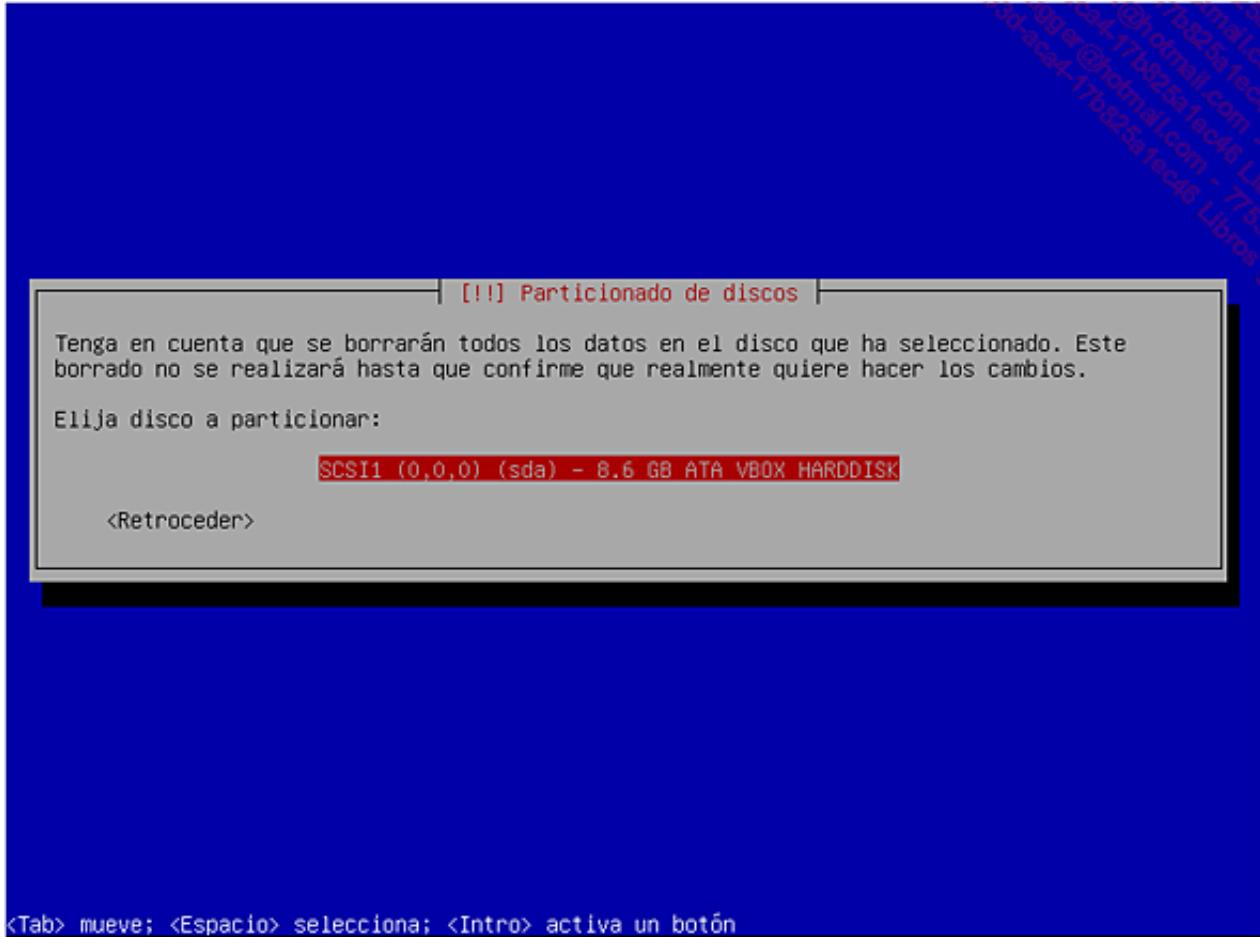
---



<Tab> mueve; <Espacio> selecciona; <Intro> activa un botón

#### *Etapa de particionado*

Si ha elegido el primer método, accede a la pantalla siguiente. Para llevar a cabo las tareas propuestas en este libro, se ha creado un espacio de 8 GB (más o menos) bajo VMWare como primer disco SCSI. Servirá para la instalación.

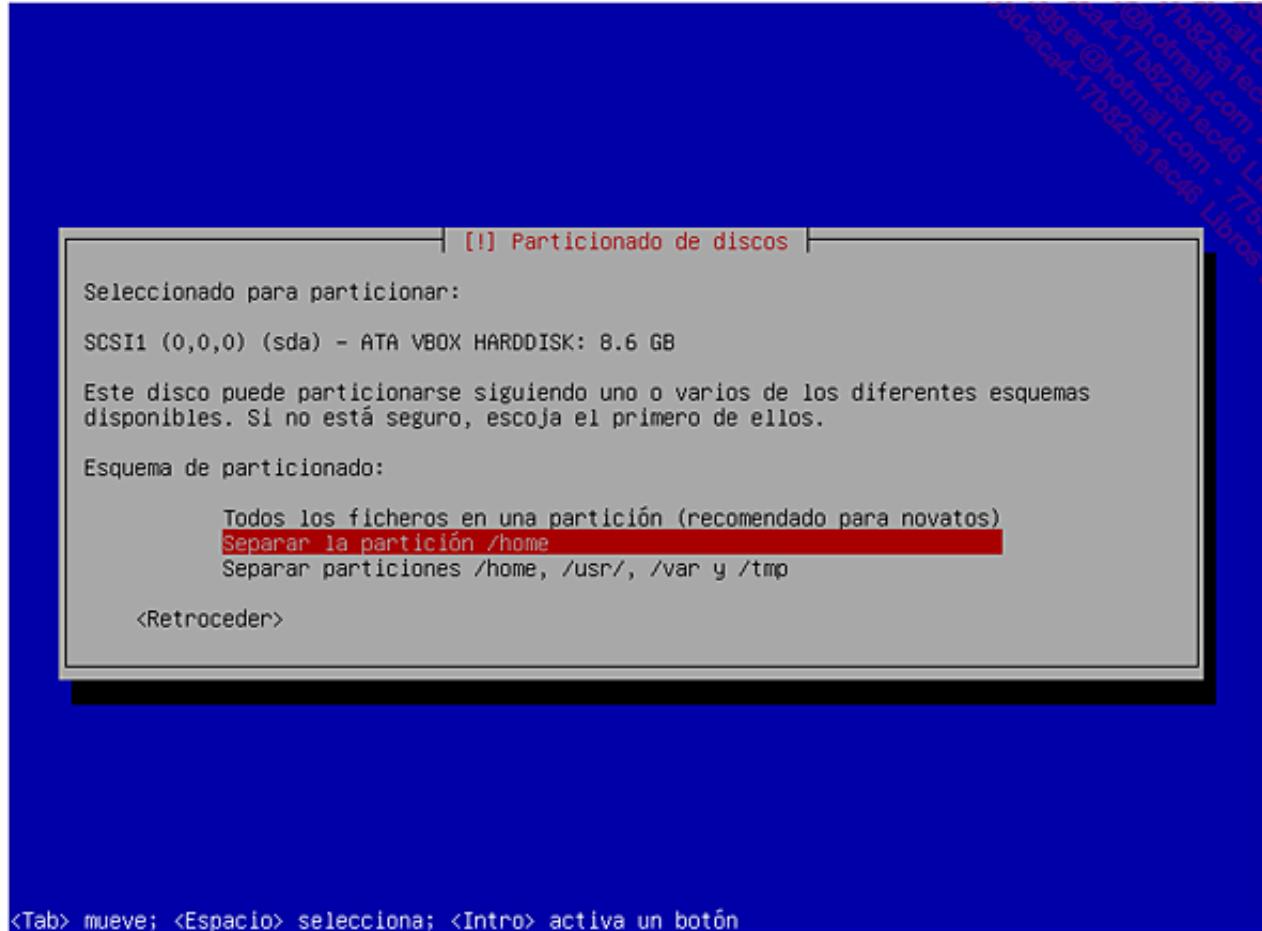


#### *Elección del disco*

La etapa siguiente consiste en elegir el esquema de particionado:

- O una sola partición grande en la cual usted lo pone todo (sistema, programas, datos). En tal caso, sólo necesita la partición raíz /.
- O dos particiones: una partición raíz que contendrá el sistema y todos sus componentes (programas, parámetros del sistema, etc.), y una partición que contendrá los datos de los usuarios. Para un puesto de trabajo o un PC personal (en casa), es el método más pertinente: permite instalar de nuevo fácilmente otro sistema (actualización o reinstalación completa) sin comprometer los datos personales. La nueva distribución instalada de esta manera podrá utilizar de nuevo la partición montada en /home y, por lo tanto, recuperar los datos.
- El tercer método propone crear cinco particiones diferentes: la raíz /, los datos personales /home, los componentes de los usuarios (programas, librerías, datos compartidos asociados, etc.) /usr, el contenido variable /var y los ficheros temporales /tmp. Esta opción es perfectamente apropiada para un servidor. Los mails, información DHCP, sitios web, etc., a menudo están almacenados en /var. Las actualizaciones de los distintos programas (servicios) están en /usr, etc. Este esquema de particionado es casi perfecto: todo es independiente. De este modo, resulta mucho más sencillo cambiar el disco, migrar los datos, extender el tamaño de los volúmenes, etc., sin comprometer el resto.

La mejor opción para la instalación de prueba es la segunda, y por esa razón la ha utilizado este manual.



#### *Eleción del esquema de particionado*

Al tratarse de un método semiautomatizado, Debian le indica sus opciones para cada una de las particiones. Puede constatar la presencia de tres particiones en lugar de las dos propuestas. Debian ha analizado el tamaño de la máquina y propone la creación de una zona de swap que corresponde a la mejor opción posible.

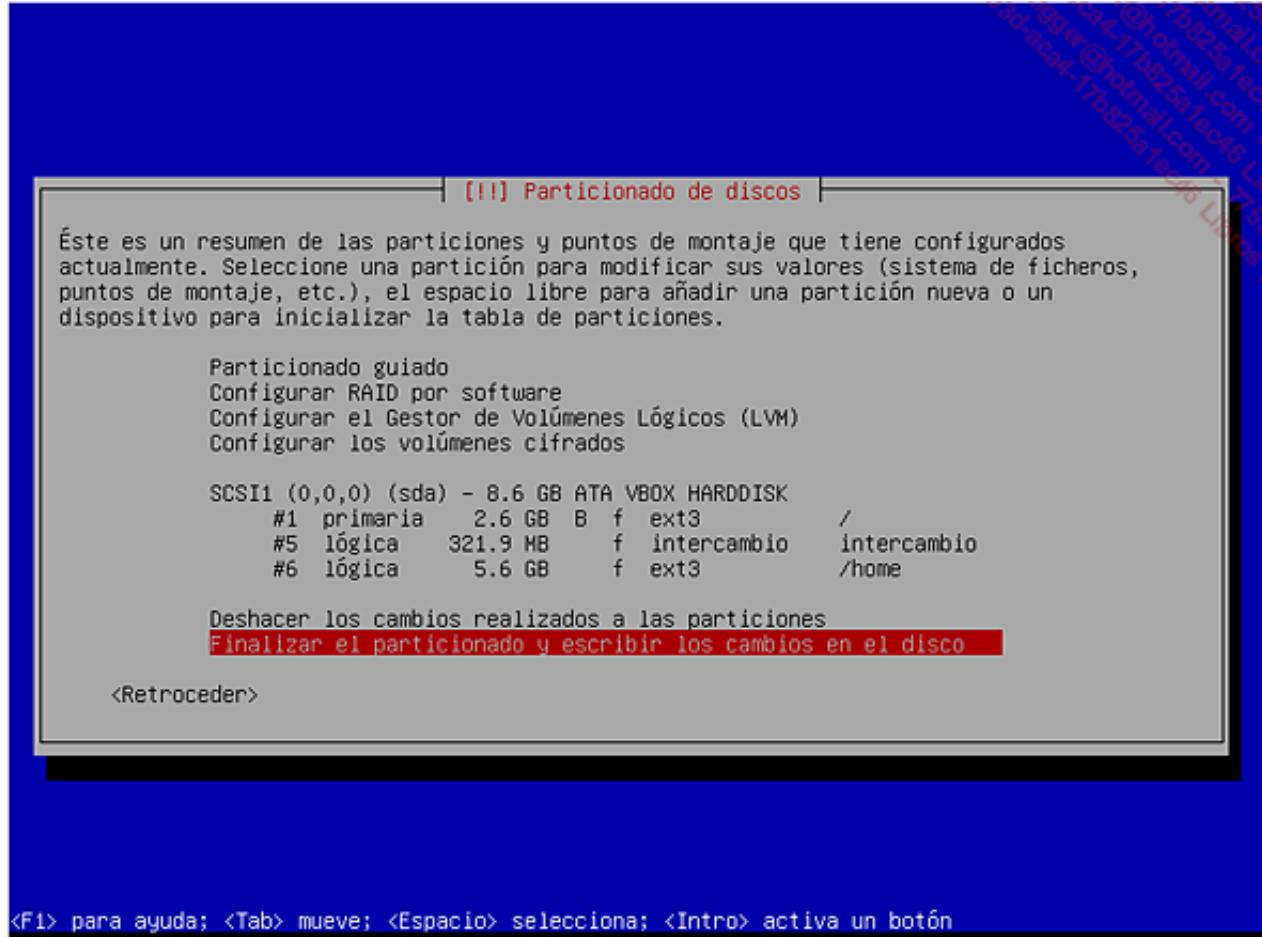
No se sorprenda tampoco por la cantidad de particiones. Muy a menudo, la única partición primaria es la raíz /, mientras que todas las otras son particiones lógicas dentro de una partición extendida, y Linux numera las primarias de 1 a 4 (una partición extendida es una partición primaria), mientras que la numeración de las particiones lógicas empieza con el 5. Para una mejor comprensión, véase el capítulo Los discos y el sistema de ficheros.

Si este esquema de particionado le conviene, valide. La pantalla siguiente muestra un resumen que debe de nuevo validar.

---

 ¡Cuidado! El particionado va seguido de la escritura de los sistemas de ficheros en las particiones correspondientes. Esta operación es idéntica al la de formatear en Windows y supone la destrucción de la información de la partición.

---



#### *Confirmación definitiva del particionado*

Una vez validados los cambios, una barra de progreso le informa del estado del particionado y de la escritura de los nuevos sistemas de ficheros. Debian montará después estos sistemas para la instalación.

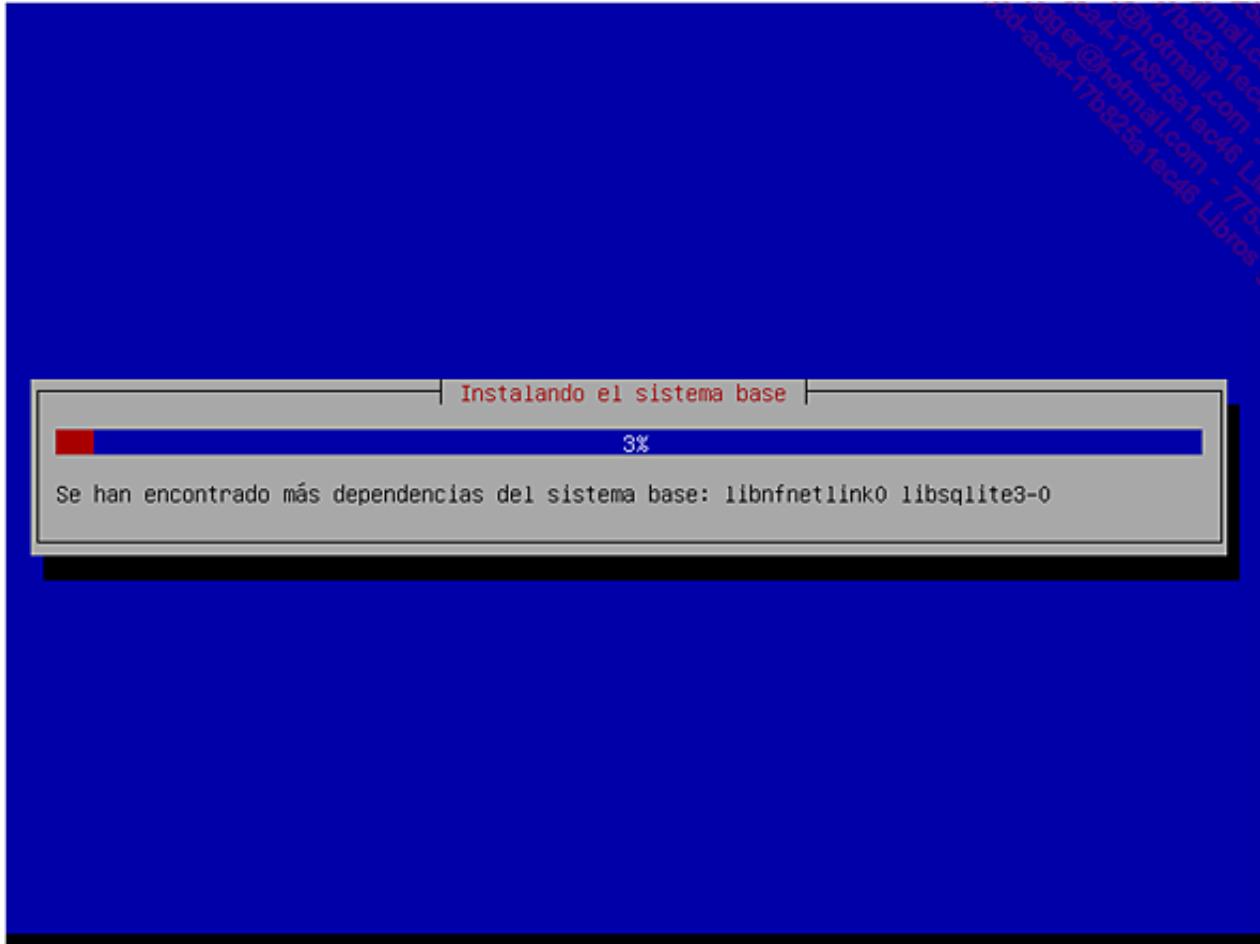
## 7. Instalación

El soporte de instalación elegido contiene lo necesario para instalar un sistema básico en su disco. En esta etapa, automática, se instalarán todos los paquetes básicos (núcleo, bibliotecas, shell, etc). A continuación, si no desea instalar paquetes, tendrá un sistema minimalista pero funcional. Es en este momento y en la etapa de la elección de un servidor réplica del repositorio de paquetes que la configuración de red tendrá lugar.

Si el instalador no ha conseguido configurar la tarjeta de red con DHCP, solicitará la introducción de los datos básicos: dirección IP, máscara de subred, puerta de enlace predeterminada y DNS.

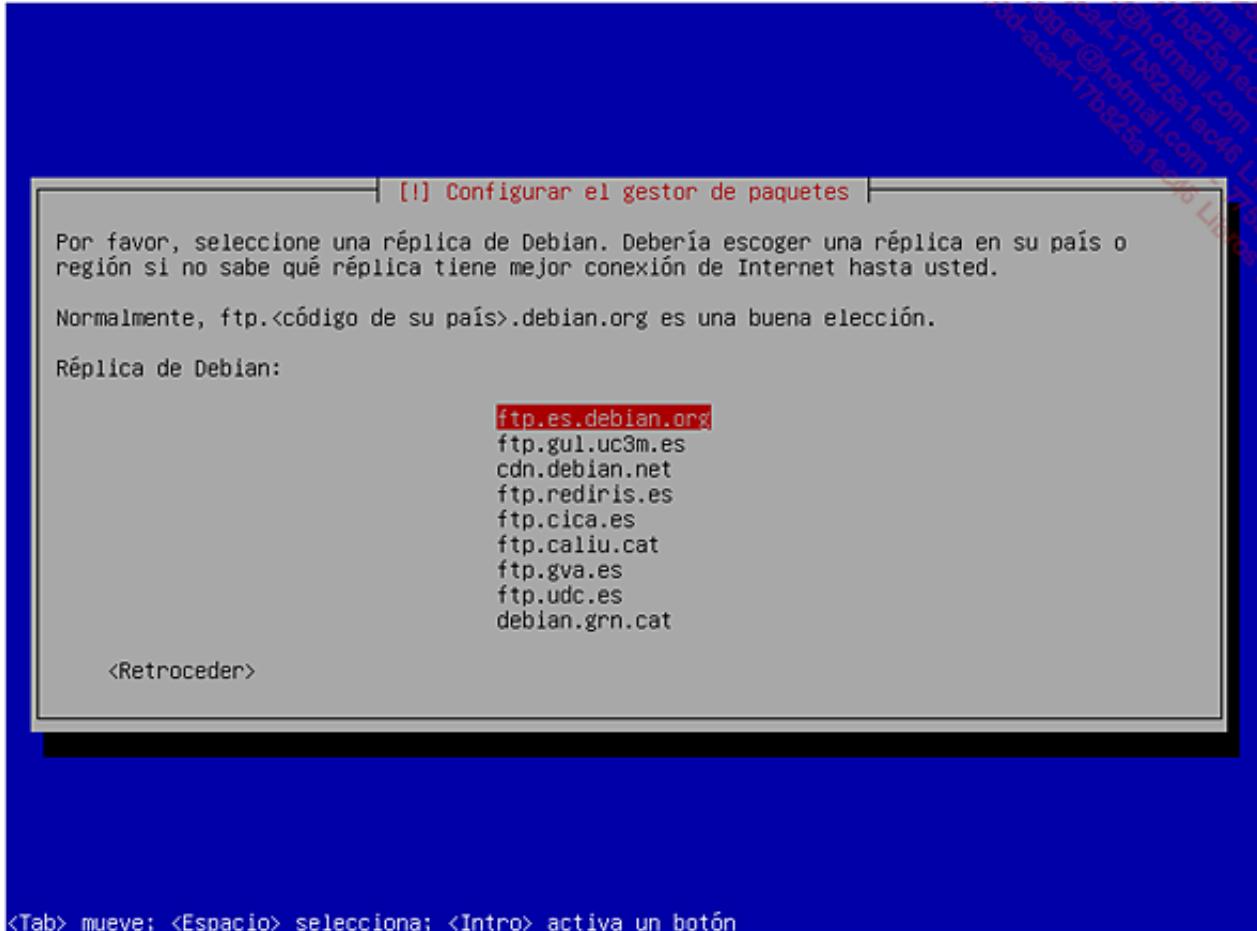
La instalación se desarrolla en varias etapas:

- Instalación de los elementos básicos. Ya se trate de elementos descargados o contenidos en el soporte de instalación, se vuelven a copiar los elementos básicos en su disco. Se trata de los paquetes principales.



### *Instalación del sistema básico*

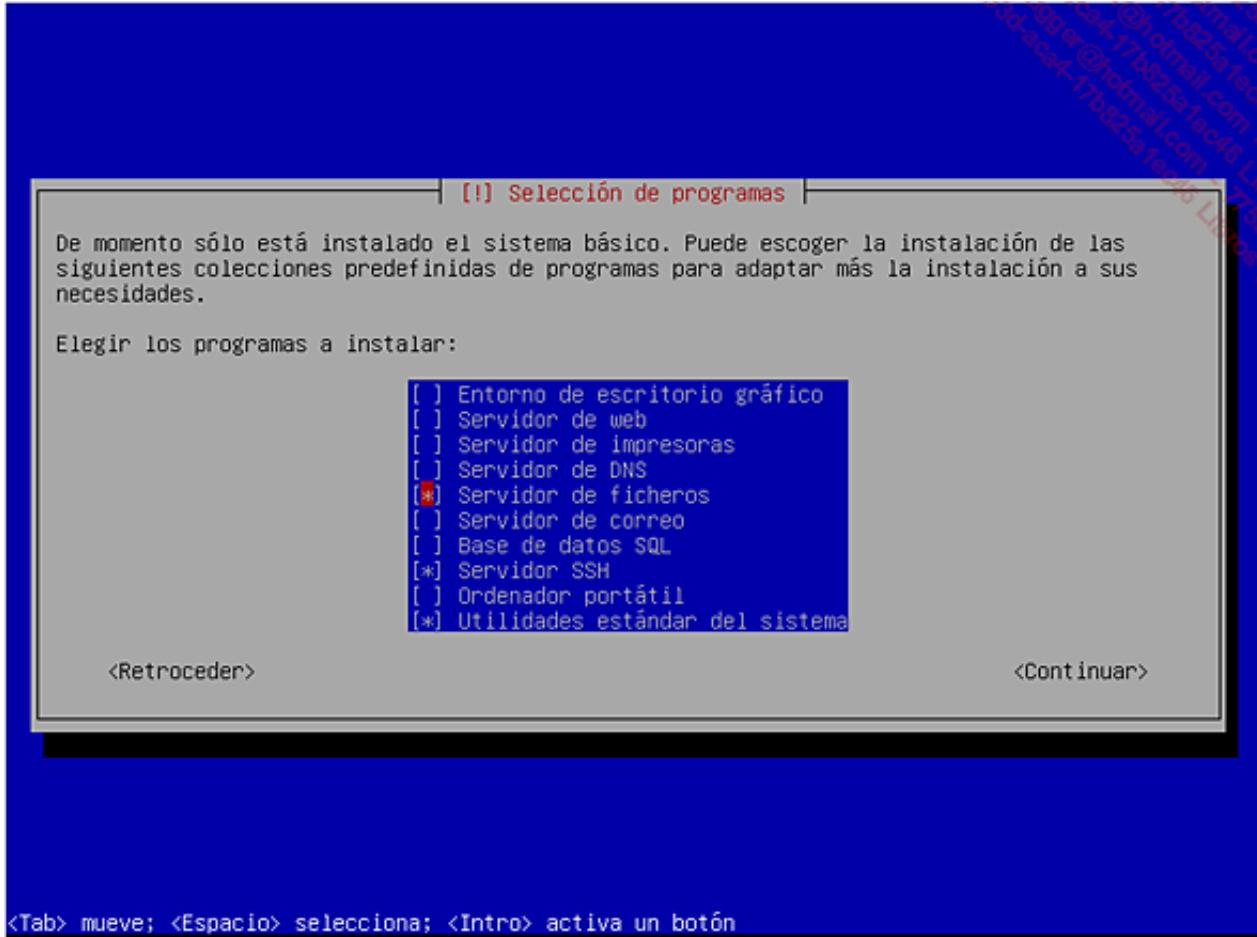
- Luego elija una réplica (mirror) de Debian: es el lugar (en Internet) donde está depositado todo el software (repositorios) Debian. Si no utiliza repositorios, Debian usa únicamente el contenido del soporte de instalación. Si se trata de un DVD, no hay problemas, pero si utiliza, como aquí, un CD de instalación en red, sólo dispone de la base y nada más. Las réplicas contienen también las actualizaciones, en particular de seguridad, aparecidas entre el momento en el cual salió el software y el momento de la instalación. Se aconseja encarecidamente que configure uno de estos repositorios aunque disponga de un juego de soportes. En caso contrario, tendrá instalado un sistema básico.



<Tab> mueve; <Espacio> selecciona; <Intro> activa un botón

#### *Eleción de la réplica de los repositorios Debian*

- Más adelante debe elegir entre uno o varios conjuntos de programas predefinidos. Estos conjuntos se agrupan por temas: entornos de escritorio (kde, gnome, xfce, etc.), los diferentes servidores (ficheros, impresión, dns, correo, SQL, etc.).

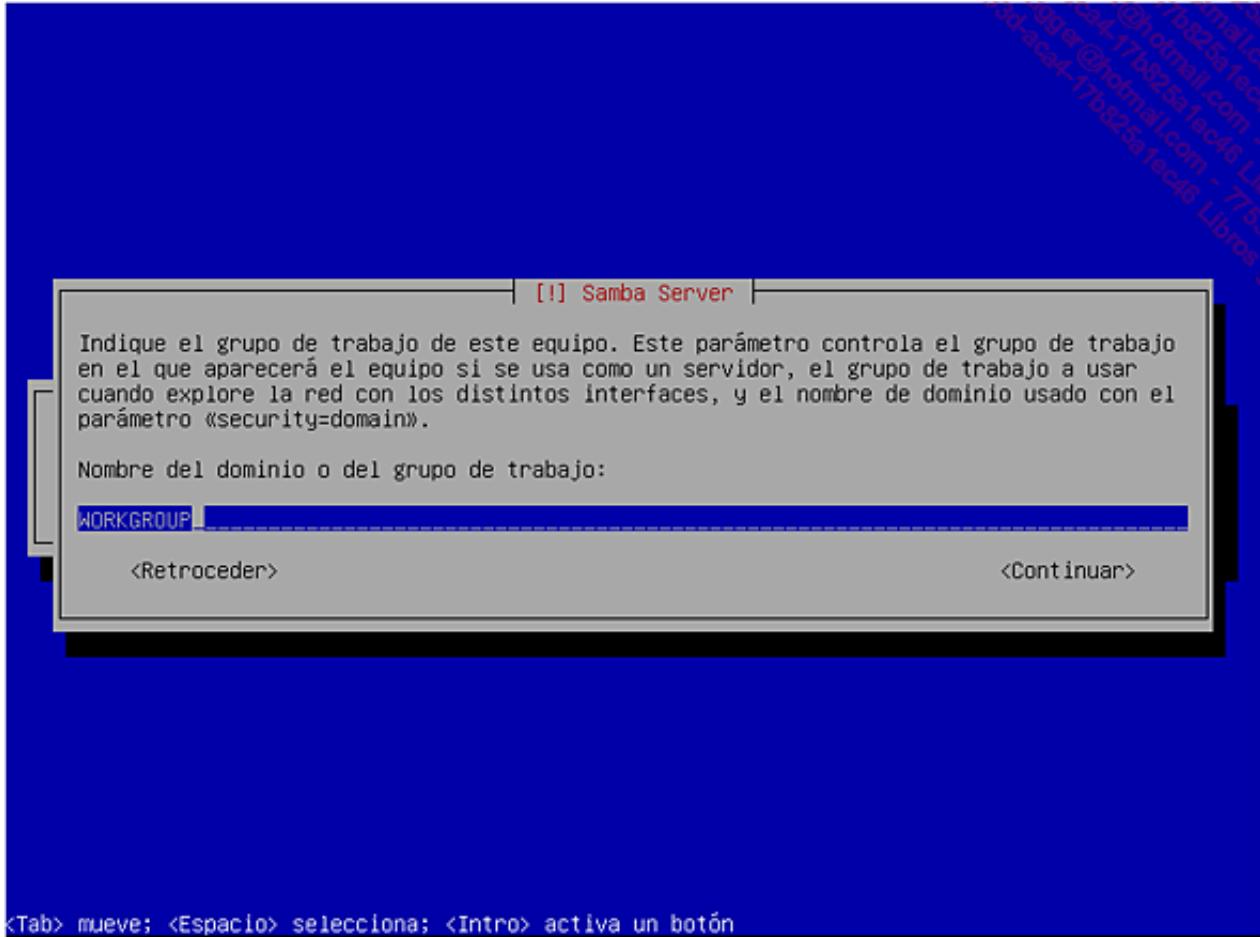


*Elección de los elementos predefinidos a instalar*

Finalizada esta etapa de configuraciones, Debian comienza su instalación.

## 8. Configuración de los paquetes

Durante la instalación, puede que deba configurar algunos paquetes. Es el caso de, por ejemplo, el servidor Samba (particiones para Windows), donde se le pedirá el tipo servidor, el dominio, etc. Sucedería lo mismo con la configuración del servidor gráfico. No es posible darle una lista de cada una de las etapas de configuración de estos paquetes. Sin embargo, en caso de errores en la configuración es posible volver a reconfigurar el paquete correspondiente (se indica su nombre arriba; en el ejemplo se trata de `xserver-xorg`) usando el comando **dpkg-reconfigure** (p. ej.: `dpkg-reconfigure xserver-xorg`).



<Tab> mueve; <Espacio> selecciona; <Intro> activa un botón

*Ejemplo de instalación del servidor SAMBA*

## 9. Fin de instalación y reinicio

La última etapa antes del final de la instalación es la escritura del cargador de arranque (bootloader). Se trata de la instalación de **GRUB**. Es muy sencillo en este caso, ya que es el único sistema instalado. Pero debe saber que los cargadores como Grub invalidará el instalado anteriormente. Puede utilizar perfectamente grub para arrancar con cualquier sistema, incluido Windows.

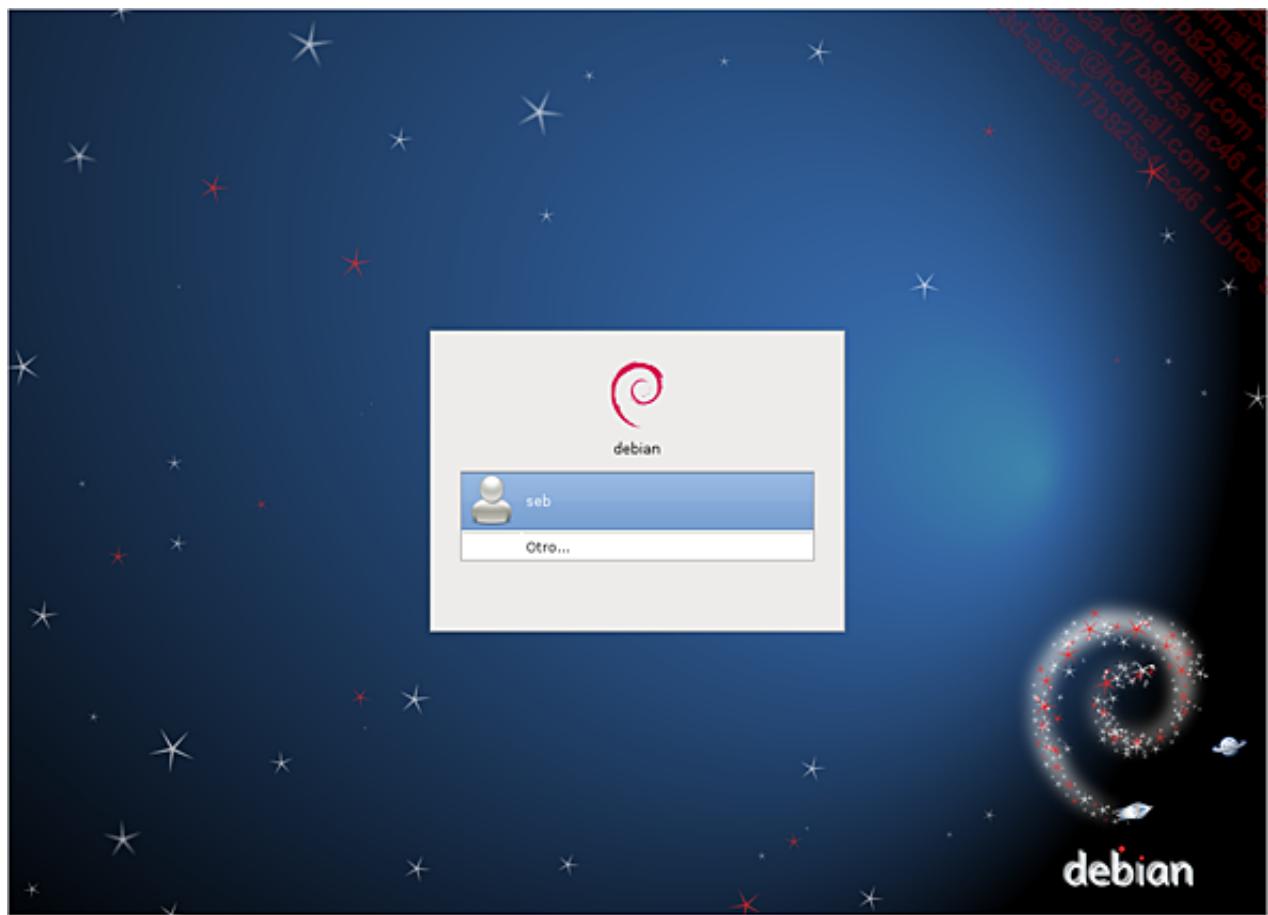


*Arranque con GRUB*

Ya sólo falta volver a «bootear». Grub le propone por defecto dos entradas:

- el modo normal;
  - el modo single-user o modo de emergencia.

Inicie en el primero. Si todo va bien y se reconoce la tarjeta gráfica, debe llegar al gestor de sesiones. Cabe destacar que puede modificar el tema visual a su gusto, las elecciones predeterminadas de Debian no son necesariamente las mejores.



Todo marcha correctamente. El gestor de sesiones está a la espera de una conexión.

# Instalación de Fedora

## 1. Soporte de instalación

A continuación se muestra el proceso de instalación paso a paso de una distribución basada en paquetes en formato RPM con un instalador en modo gráfico. Siendo menos sobria que la distribución Debian, que es más simple pero más extendida, la instalación de una distribución Fedora se acerca a la de una distribución Red Hat en el ámbito empresarial.

La instalación se ha realizado en una máquina virtual Virtualbox configurada del mismo modo que con la distribución Debian, a excepción de la elección de Fedora 64 bits en el asistente de configuración. Si desea instalar la misma distribución, se puede acceder a través de la URL [http://ftp.cica.es/fedora/linux/releases/15/Fedora/x86\\_64/iso/Fedora-15-x86\\_64-DVD.iso](http://ftp.cica.es/fedora/linux/releases/15/Fedora/x86_64/iso/Fedora-15-x86_64-DVD.iso)

La instalación se ha realizado desde un DVD de la versión Fedora 15 de 64 bits. También se puede realizar la instalación a través de la red, del mismo modo que con Debian.

## 2. Arranque del soporte

Configure su ordenador para que arranque a través del soporte e inícielo.



Pantalla de inicio del soporte de instalación Fedora

La pantalla de inicio del soporte de instalación ofrece una gran cantidad de opciones y es menos sobria que

otras distribuciones actuales. Las dos primeras opciones permiten instalar o actualizar el sistema, la segunda utilizando un driver básico compatible con cualquier tarjeta gráfica. Esta segunda opción además permite iniciar desde una copia de seguridad del sistema que acompaña el soporte de instalación. De este modo, se puede reparar su sistema incluso si es imposible iniciar desde el disco duro. El arranque con el disco local redirige el arranque a su disco duro.

Una opción curiosa es la comprobación de la memoria. Muchas disfunciones de un equipo se asocian a menudo erróneamente al sistema operativo: pantallas de errores (las famosas BSOD, aunque el color haya cambiado en las últimas versiones) de Windows o kernel panic en Linux. Puede que incluso Linux sea inestable en algunos PC. Ahora bien, con frecuencia el fallo del sistema se debe a un problema de hardware, de los cuales el más común es un problema en uno o varios módulos de memoria. La utilidad memtest+ permite realizar un diagnóstico de dichos módulos mediante pruebas pesadas (y de larga duración). Lo ideal ante un cuelgue inesperado es iniciar este test conservando un solo módulo e ir probando los módulos uno tras otro.

---

 En informática, como en otros ámbitos, la calidad tiene un coste generalmente elevado. Evite componentes de marca "blanca" o genéricos y escoja preferiblemente una marca conocida para todos sus componentes, sobre todo para la memoria (Corsair, Gskill, Kingston, PQI, etc.), las placas base (Gigabyte, Asus, DFI, etc.) y las fuentes de alimentación (Emermax, Fortron, etc.). Asegúrese de la compatibilidad entre sus componentes a través de los vendedores y de la documentación técnica. El autor ha visto módulos de memoria de gran calidad que no funcionan en ciertas placas base y una webcam incluso inactiva bloqueando una tarjeta gráfica cuando se usaba en 3D...

---

Seleccione la primera opción y pulse [Intro].

---

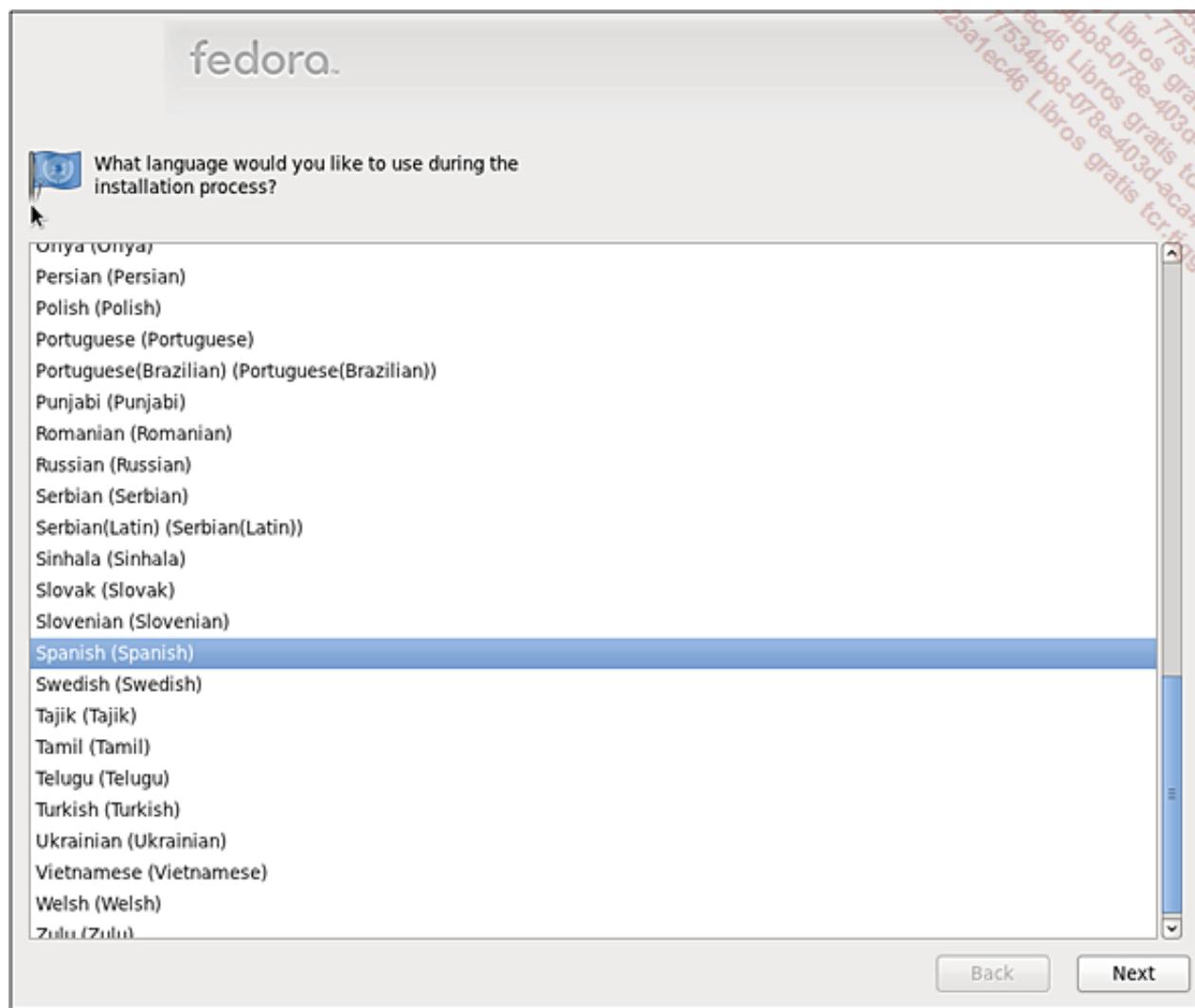
 Cabe destacar que puede intercambiar la pantalla en modo texto pulsando la tecla [Esc]. En este caso se encontrará con la clásica pantalla de tipo Grub muy parecida a la de la instalación Debian.

---

Fedora le solicitará si debe comprobar el soporte de instalación. Esta decisión depende completamente de usted. Para evitarlo, colóquese en **skip** usando la tecla de tabulación y pulse [Intro]. Anaconda, la herramienta de instalación común de Fedora, Red Hat y CentOS, se iniciará.

### 3. Idioma de la instalación

La herramienta de instalación **Anaconda** es una rara constante, debido a que la interfaz apenas ha evolucionado en los últimos diez años. Esta interfaz puede funcionar en modo texto (consola) o en modo gráfico, que es el que se ha usado en este libro.



### *Idioma predeterminado de Fedora*

Debe seleccionar el idioma de instalación, en este caso **Spanish (Spanish)** y hacer clic en **Next**. A partir de este momento, la interfaz se traducirá al castellano. La siguiente etapa consiste en elegir el tipo de teclado y, como en Debian, el español se selecciona con **Español**. A continuación, haga clic en **Siguiente**.

## 4. Dispositivos de almacenamiento destino

La siguiente etapa puede sorprender a un neófito pero no al administrador de sistemas de una empresa. Como Fedora sirve de base para las distribuciones Red Hat profesionales, ofrece elecciones relativas a este contexto.

fedora.

¿Qué tipos de dispositivos están involucrados en su instalación?

**Dispositivos de almacenamiento básicos**

- Instala o actualiza tomando en cuenta tipos de dispositivos de almacenamiento clásicos. Si no conoce exactamente las características de sus dispositivos, probablemente ésta sea la opción correcta.

**Dispositivos de almacenamiento especializados**

- Instala o actualiza dispositivos de tipos empresariales como son las Storage Area Networks (SANS, redes de área de almacenamiento). Esta opción le permite agregar discos FCoE / iSCSI / zFCP, y filtrar dispositivos de salida que el instalador debería ignorar.

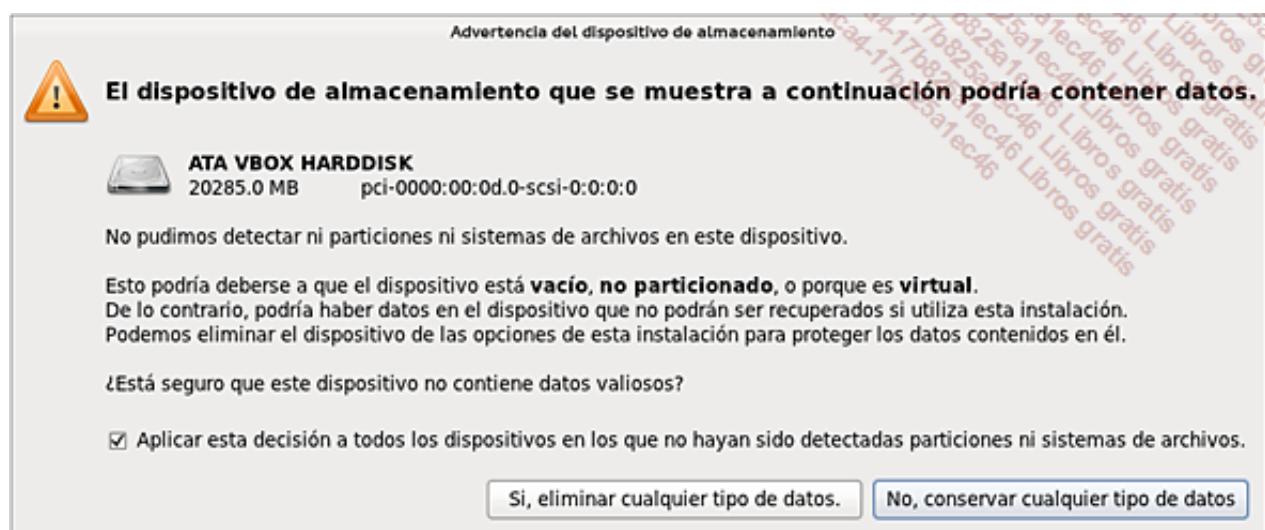
Atrás

Siguiente

*Fedora es compatible con dispositivos de almacenamiento empresariales*

En una máquina virtual o en su PC de escritorio, es evidente determinar a qué va dirigida la primera opción: la instalación se realizará en uno de los discos conectados a su PC, principalmente un disco IDE, SATA o SCSI. La segunda opción es útil en ámbitos empresariales donde el sistema se puede instalar en un disco SAN o iSCSI. Haga clic en **Siguiente**.

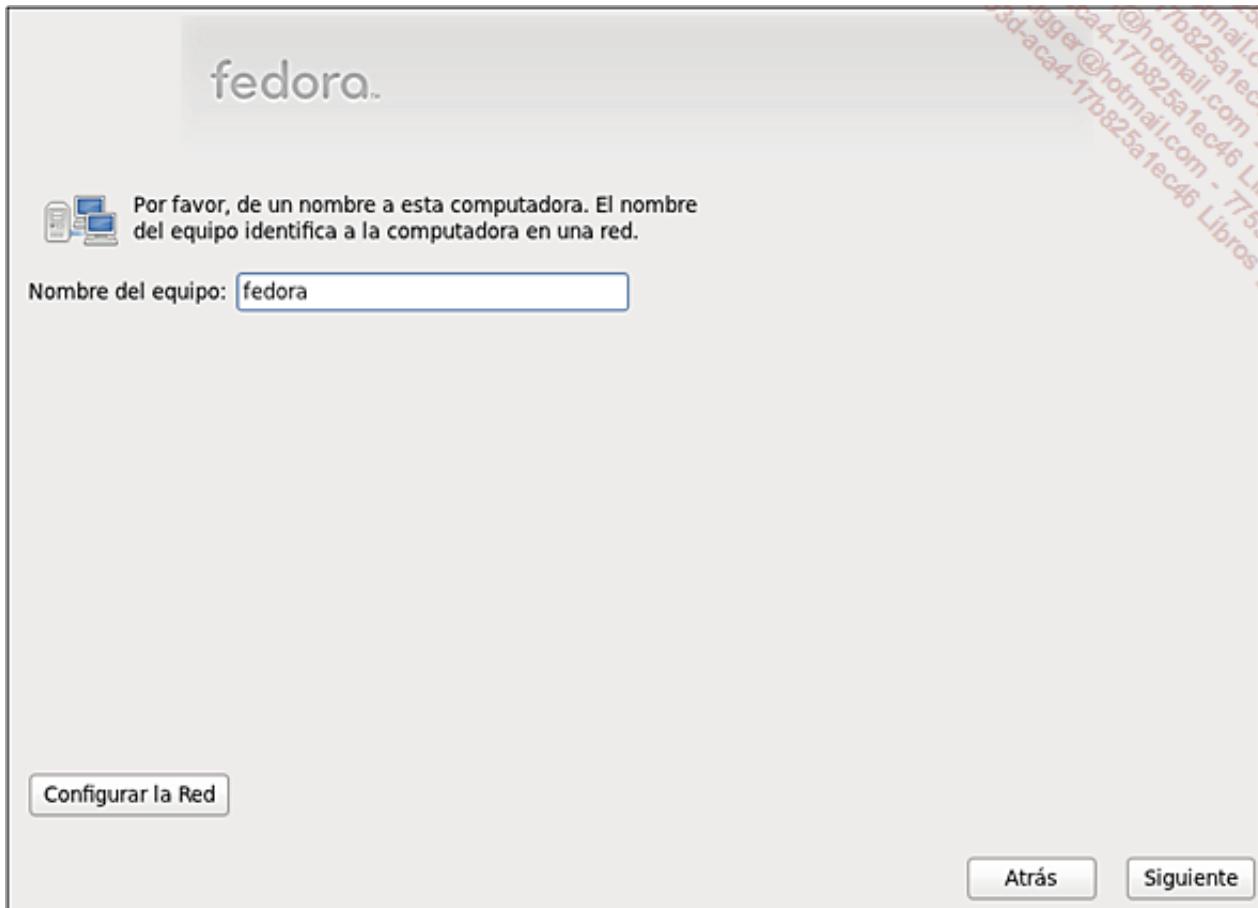
Anaconda detecta los discos en los que se podría instalar. Si detecta un disco cuyo contenido no consigue reconocer, podrá aparecer el siguiente cuadro de diálogo. ¡Ponga especial atención en su respuesta! ¡Algunos discos no particionados pueden contener volúmenes físicos LVM o sistemas de archivos formateados! Verifique su elección asegurándose de que su disco está completamente vacío. Tenga cuidado con la opción seleccionada por defecto. Si usted la deja seleccionada, su elección se aplicará a todos los discos en ese mismo estado.



*¿Está el disco vacío?*

## 5. Configuración de la red

La siguiente etapa permite elegir el nombre de anfitrión y, si se considera oportuno, con el botón **Configurar la Red**, modificar los parámetros asociados. Es aquí donde podrá configurar sus interfaces de red de forma estática o por DHCP.



*Configuración del nombre de anfitrión*

Puede configurar cualquier tipo de conexión (cable, Wi-Fi, ADSL, etc.). Si, por ejemplo, debe modificar su adaptador de red Ethernet, haga clic en el botón **Configurar la Red**. En el cuadro de diálogo **Conexiones de red**, compruebe que está en la pestaña **Cableado**, haga clic en su interfaz y a continuación pulse **Editar**. Vaya a la pestaña deseada, por ejemplo **Ajustes de IPv4**, y modifique sus datos IP (puerta de enlace estática, introducción de sus direcciones IP, etc.).

**Editando System p2p1**

Nombre de la conexión: **System p2p1**

Conectar automáticamente

Cableado	Seguridad 802.1x	Ajustes de IPv4	Ajustes de IPv6
----------	------------------	-----------------	-----------------

Método: **Automático (DHCP)**

### Dirección

Dirección	Máscara de red	Puerta de enlace	Add
			Delete

Servidores DNS:

Dominios de búsqueda:

ID del cliente DHCP:

Requiere dirección IPv4 para que esta conexión se complete

---

Disponible para todos los usuarios

## *Configuración avanzada de una interfaz de red*

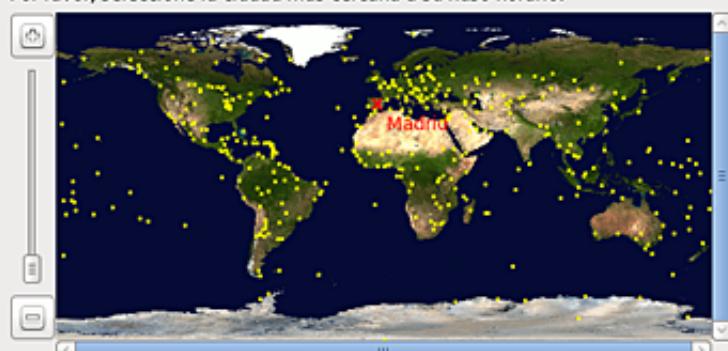
Haga clic en **Siguiente** una vez haya finalizado el procedimiento de configuración.

## 6. Reloj

Seleccione el huso horario correspondiente a su país o ciudad.

fedora.

Por favor, seleccione la ciudad más cercana a su huso horario:



Ciudad seleccionada: Madrid, Europa (área principal)

Helsinki

Isla del Hombre

Jersey

Kaliningrado

Moscú-01 - Kaliningrado

Kiev

la mayoría de las localidades

Lisboa

área principal

Ljubljana

Londres

Luxemburgo

Madrid

Use esto si quiere cambiar automáticamente entre momentos normales y de ahorro de luz del día. No lo use si ya tiene otros sistemas operativos en esta computadora que ajustan el hardware de reloj para hacer esto, por ejemplo Microsoft Windows (tm).

El reloj del sistema utiliza UTC

Atrás

Siguiente

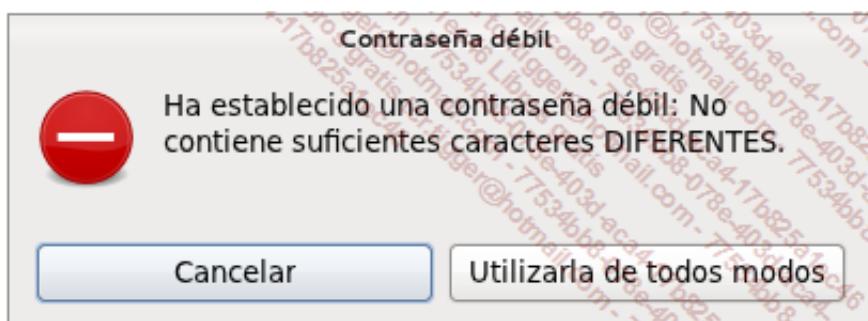
### ¿Huso horario y reloj en UTC?

Si el reloj de su máquina está realmente en hora, es decir, ajustado a la hora exacta del huso horario seleccionado, desmarque **El reloj del sistema utiliza UTC**. En caso contrario, Linux considerará la hora del sistema como UTC y calculará la hora real en función del decalaje asociado al huso horario, por ejemplo UTC +1 en invierno y UTC+2 en verano.

Haga clic en **Siguiente**.

## 7. Usuario root

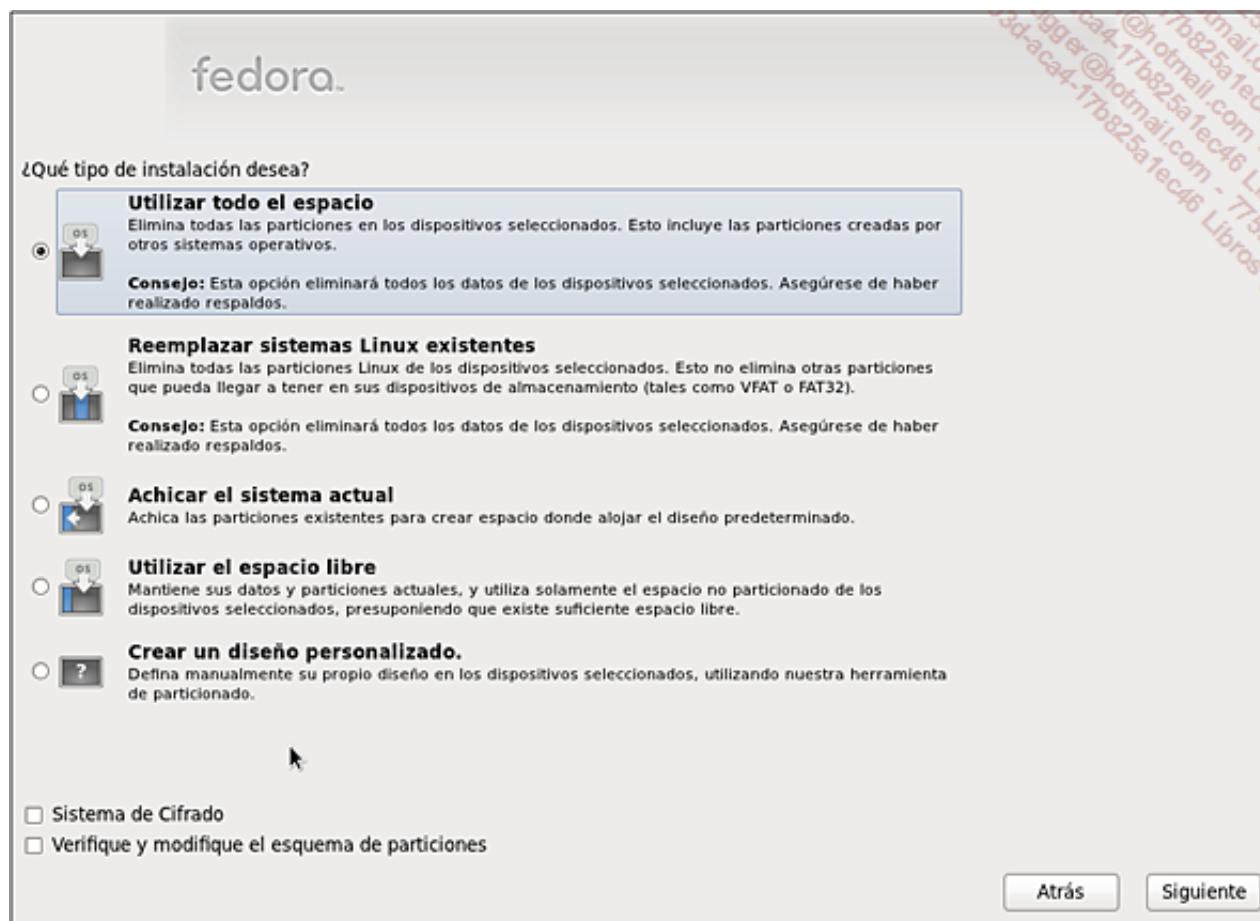
Introduzca la contraseña del usuario root, administrador del sistema. Se evaluará la complejidad de la contraseña y si es demasiado sencilla Anaconda mostrará un mensaje de error. A pesar de ello, podrá continuar si lo desea.



Haga clic en **Siguiente**.

## 8. Particionado

Como con Debian, Fedora le ofrece varias posibilidades de cara al particionado de su disco. La primera opción consiste en utilizar todo el disco. Anaconda gestiona muchas opciones. Tiene la posibilidad, por ejemplo, de reutilizar las particiones Linux existentes borrando las versiones ya existentes, reducir el tamaño y los sistemas de archivos ya presentes (especialmente las particiones Windows en NTFS), sólo utilizar el espacio libre o particionar el disco usted mismo.



*Fedora ofrece multitud de esquemas de particionamiento*

Cabe destacar las dos casillas de selección de abajo. Si decide cifrar el sistema, las particiones y todos los datos se encriptarán y, por tanto, serán ilegibles desde otro sistema operativo, incluyendo Linux. Si elige esta opción, sobre todo no olvide los datos de autenticación que proporcione, o correrá el riesgo de no poder acceder a sus propios datos, especialmente si reinstala o actualiza su distribución.

La segunda casilla de selección permite particionar después de haber preconfigurado las acciones seleccionadas. Active esta casilla.

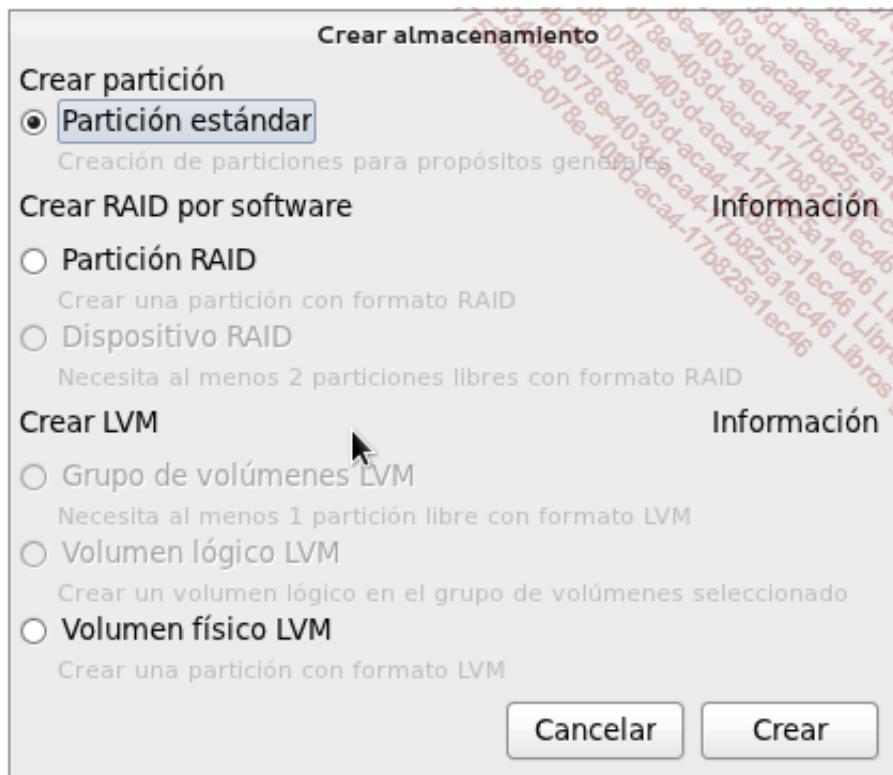
Las elecciones de particionado predeterminadas reflejan la vocación profesional de Red Hat. Anaconda ofrece de forma predeterminada establecer una configuración de tipo LVM. Esto es muy práctico y muy flexible, pero poco sencillo si acaba de empezar a introducirse en el mundo Linux y está habituado a un esquema de partición/sistema de archivos clásico. Por lo tanto, puede modificar la tabla de particiones.



*Fedora prefiere LVM por defecto*

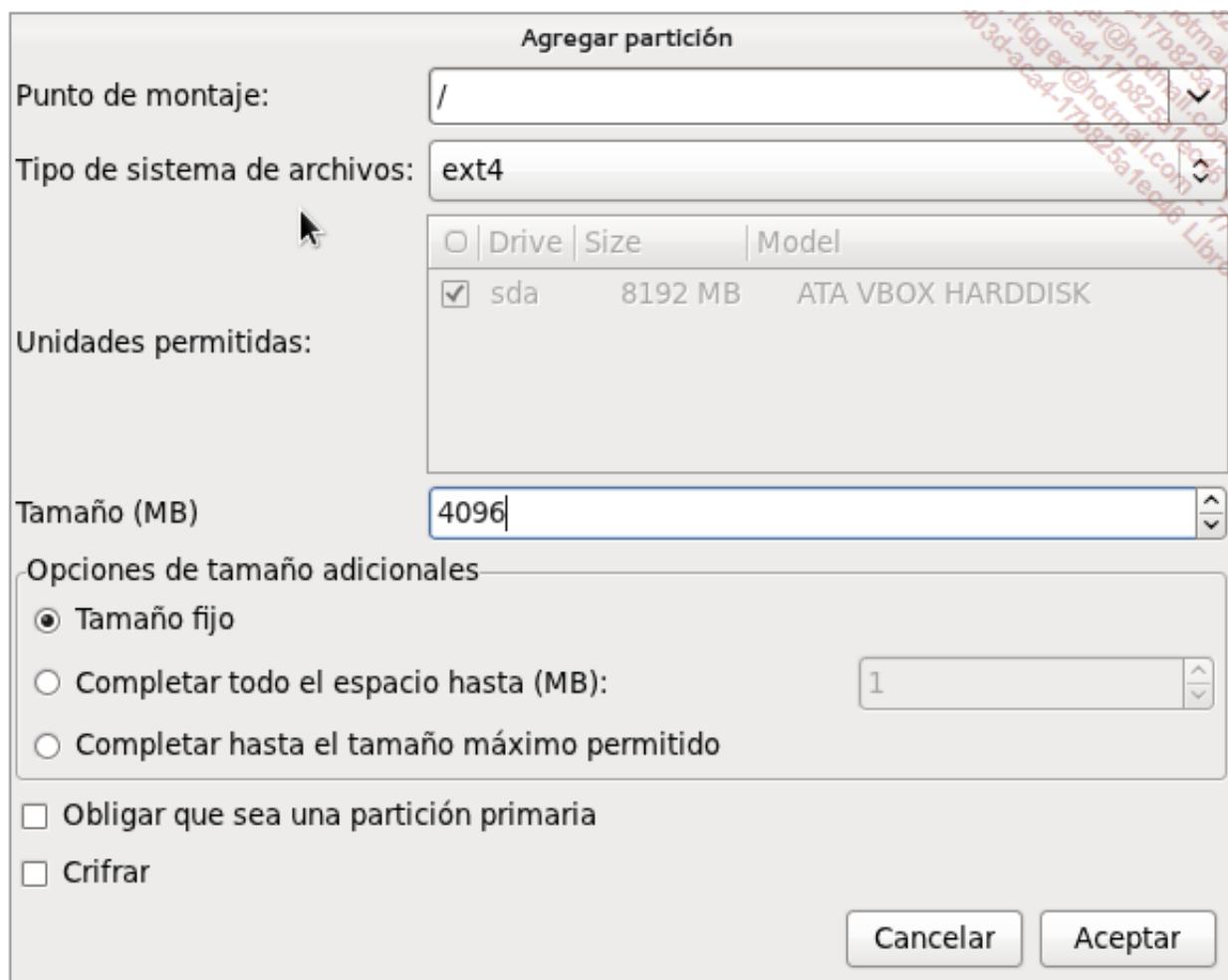
A continuación se explica cómo crear uno mismo las particiones necesarias:

- Haga clic en **Reiniciar**, lo que vaciará la tabla y a continuación haga clic en **Crear**.
- En la pantalla de creación del espacio de almacenamiento, seleccione **Partición estándar**, y a continuación haga clic en **Crear**.



### Paso 1: crear una partición estándar

- Elija el punto de montaje raíz “/”, asígnele un tamaño de alrededor de 4096 MB y haga clic en **Aceptar**.



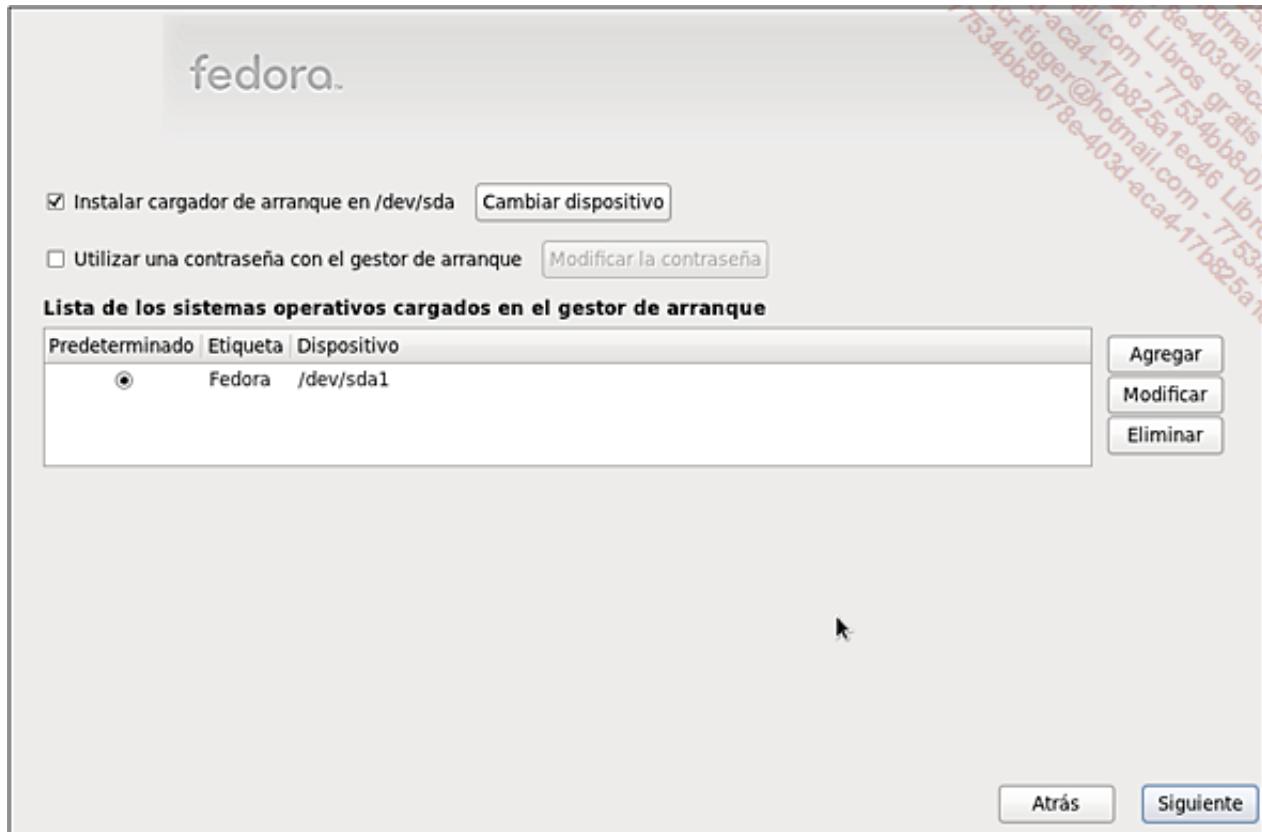
### Paso 2: tipo, punto de montaje y tamaño

- Haga lo mismo para la partición /home, sin olvidar que hay que reservar el espacio para una partición de swap.
- Cree a continuación una partición de swap con el tamaño restante, eligiendo swap como tipo de sistema de archivos y rellenando el disco hasta el tamaño máximo permitido.

Dispositivo	Tamaño (MB)	Punto de Montaje/ RAID/Volumen	Tipo	Formato
▼ Discos duros				
▼ sda (/dev/sda)				
sda1	4096	/	ext4	✓
sda2	3500	/home	ext4	✓
sda3	595		swap	✓

Esquema final

- Acepte y confirme sus elecciones. Las particiones se crearán y los sistemas de archivos se formatearán.
- Confirme finalmente la instalación del cargador de arranque en el MBR del disco y las entradas de Grub.



*Instalación de Grub*

## 9. Selección e instalación de paquetes

La elección de paquetes queda bastante resumida por defecto. Haga clic en **Escritorio gráfico**. En esta etapa se pueden añadir repositorios adicionales y se puede personalizar la selección de los paquetes que se instalarán. Sin embargo, estas acciones pueden realizarse una vez que el sistema haya sido instalado.

fedora.

La instalación predeterminada de Fedora ofrece un grupo de aplicaciones para el uso general de Internet. Opcionalmente puede seleccionar ahora un grupo de aplicaciones diferente.

- Escritorio gráfico
- Desarrollo de software
- Servidor web
- Mínima

Por favor, seleccione repositorios adicionales que quiera utilizar para la instalación de software.

- Installation Repo
- Fedora 15 - x86\_64
- Fedora 15 - x86\_64 - Test Updates

[Añadir otro repositorio](#)

[Modificar repositorio](#)

La selección de software puede personalizarse ahora o al finalizar la instalación, utilizando el sistema de administración de software.

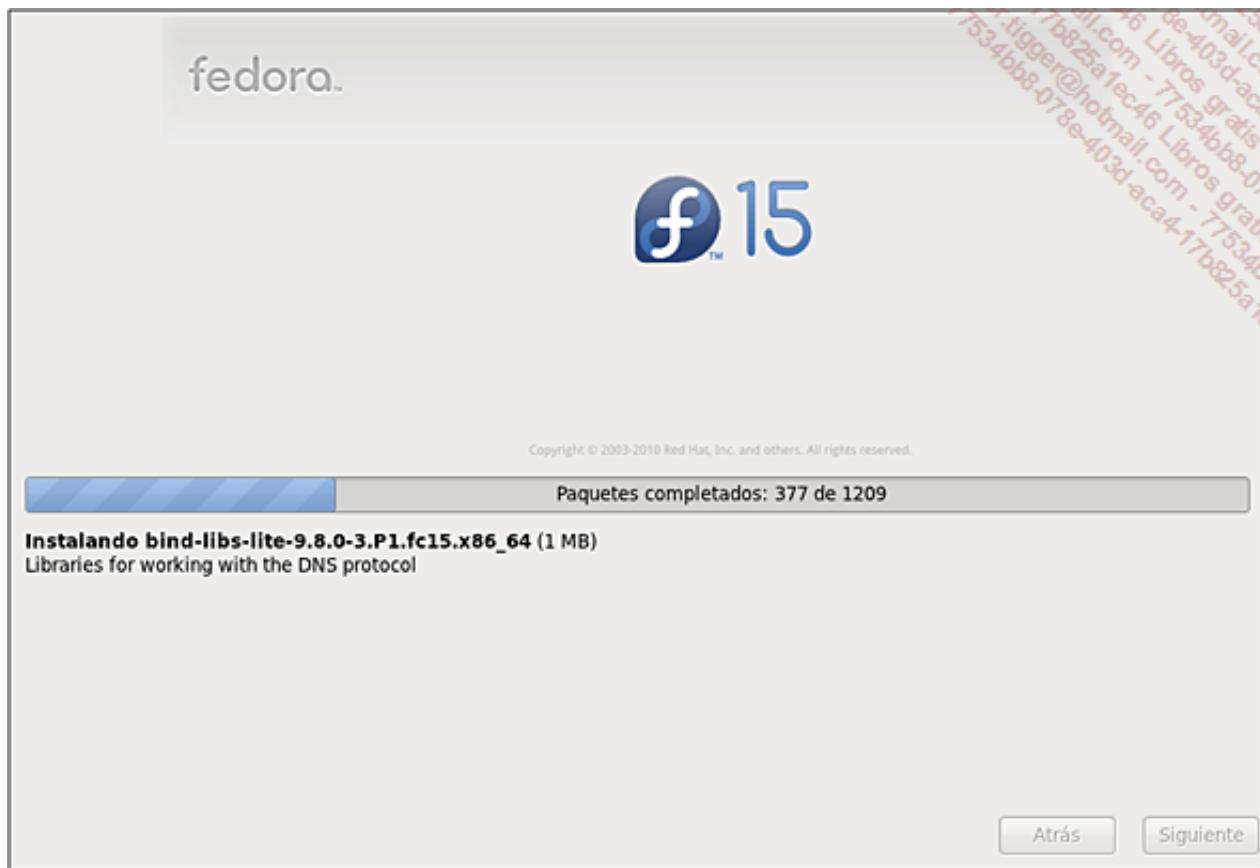
- Personalizar luego
- Personalizar ahora

[Atrás](#)

[Siguiente](#)

#### *Componentes que se instalarán y personalización*

Haga clic en **Siguiente**. El sistema se instalará. Este proceso puede durar cierto tiempo.



*La instalación está en curso*

La instalación finaliza con la configuración de postinstalación y la instalación del cargador de arranque. Una vez ésta ha terminado, retire el soporte de instalación (CD, DVD, etc.) y haga clic en **Reiniciar**.

## 10. Primer arranque

Una vez que el sistema ha reiniciado, un asistente le permitirá terminar la configuración del sistema. Ésta consiste, después de haber aceptado la licencia, en crear un usuario, en comprobar la hora del sistema y en enviar a los desarrolladores de la distribución Fedora una copia de su perfil hardware para ayudarles a asegurar la compatibilidad de la distribución con éste y para generar estadísticas.



Bienvenido  
Información de  
Licencia

Crear Usuario  
Fecha y Hora  
Perfil de Hardware

## Crear Usuario

Se recomienda crear un 'nombre\_de\_usuario' para uso normal (no administrativo) de su sistema. Para crear un sistema 'nombre\_de\_usuario', por favor, provea la información que se pide más abajo.

Nombre Completo:

Nombre de Usuario:   Add to Administrators group

Contraseña:   Strong

Confirme la Contraseña:  

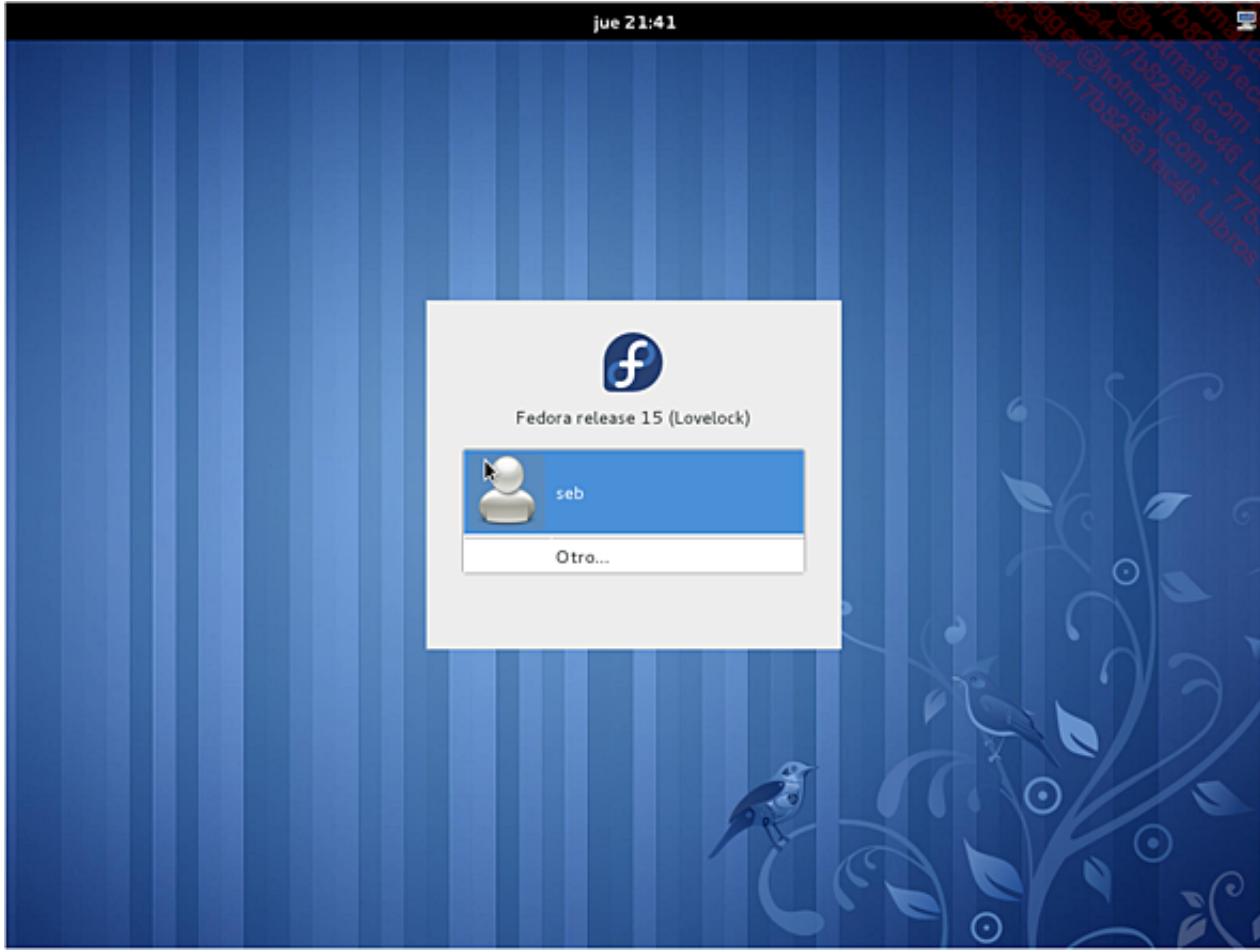
Si necesita usar autenticación de red, tal como Kerberos o NIS, por favor haga clic en el botón Usar Ingreso por Red.

Si necesita más control al crear los usuarios (especificar su directorio de inicio, y/o UID), por favor, haga clic en el botón Avanzado.



### Agregando un primer usuario

La instalación ha finalizado. A continuación se muestra el gestor de sesiones, por defecto el mismo que el de Debian, pero con otro tema.



# Red Hat Package Manager

## 1. Noción de paquete

Al contrario de otros sistemas operativos, con Linux y Unix no es habitual disponer de software proporcionado con un programa de instalación interactivo (y no install.exe). Algunos editores proponen scripts de instalación y, muy a menudo, éstos sólo descomprimen y desarchivan algunos ficheros.

Con Linux es muy habitual disponer de varios productos, herramientas, actualizaciones, etc. en forma de paquetes (packages). Un paquete es un fichero (a veces grande) que contiene el producto a instalar y unas reglas. Estas reglas pueden ser múltiples:

- Gestión de las dependencias: sólo se podrá instalar el producto si los productos que él mismo utiliza están ya presentes.
- Preinstalación: se deben prever acciones antes de poder instalar el producto (cambiar derechos, crear directorios, etc.).
- Postinstalación: se deben prever acciones después de la instalación del producto (parámetros de un fichero de configuración, compilación anexa, etc.).

En Red Hat, Fedora, SuSE, Mandriva y algunas otras distribuciones el formato de paquete por defecto es el **RPM** (*Red Hat Package Manager*). En Debian, Knoppix, Kaella, Ubuntu, es el formato **DPKG** (*Debian Package*). Además de por el formato, se diferencian principalmente por las herramientas.

El hecho de disponer de la información de dependencias permite obtener herramientas eficaces que pueden solas resolverlas en cascada. Al instalar un paquete, la herramienta podrá instalar todas las dependencias necesarias. A veces se pueden especificar varias ubicaciones (repositorios) para estos paquetes, ya sean locales (disco duro, CD-Rom, DVD, etc.) o remotos (http, ftp, etc.).

Siempre hay que utilizar un paquete previsto para su distribución cuando esté disponible. Si no es el caso, a veces es posible utilizar un paquete de un producto competidor o volver a compilar el producto por sí mismo.

Se simplifican mucho las actualizaciones de un sistema Linux que utilizan un sistema de packaging. Para pasar de la versión de un producto a la de otro, basta con recuperar el paquete del producción en versión superior e instalarlo. Todas las actualizaciones siguen este principio. Existe un formato, el **delta-rpm**, que sólo proporciona en el paquete las diferencias de una versión a otra. Pero siempre es posible utilizar un paquete completo.

## 2. El gestor RPM

**RPM** es un gestor de paquetes desarrollado por Red Hat que luego ha sido adoptado de manera masiva por muchas otras distribuciones. Simplifica mucho la distribución, la instalación, la actualización y la supresión de los programas. Se basa en comandos (p. ej.: **rpm**), una base de datos local y paquetes en formato rpm (extensión rpm).

La base de datos está colocada en `/var/lib/rpm`. Contiene toda la información correspondiente a los programas instalados, sus versiones, sus ficheros y derechos, y sus dependencias. A no ser que se produzca un grave problema, no se debe modificar NUNCA esta base de datos. Hay que utilizar las herramientas RPM.

Se facilita cada programa en forma de paquete con formato RPM. El rpm responde a una nomenclatura precisa.

nombre-versión-edición.arquitectura.rpm

por ejemplo:

`php-4.1.2-2.1.8.i586.rpm`

La edición es un identificador de versión del paquete RPM propio del editor. Aquí tenemos la versión 2.1.8 del paquete PHP versión 4.1.2. La arquitectura es i586 (Intel Pentium). También se puede encontrar i386, i686, x86\_64 (64 bits), ppc64, s390x o noarch. Un paquete noarch no contiene programas o librerías binarias, sino código independiente como scripts, documentación, imágenes, sonido, vídeo, etc.

### 3. Instalación, actualización y supresión

Se instala un paquete rpm con el parámetro `-i`.

```
rpm -i php-4.1.2-2.1.8.i586.rpm
```

Dado que es posible utilizar comodines (`rpm -i *.rpm`), puede necesitar visualizar el nombre del paquete durante la instalación; use el parámetro `-v`. El parámetro `-h` visualiza caracteres # para indicar la progresión de la instalación. La instalación no funcionará si no se resuelven las dependencias.

La actualización de un producto hacia una versión superior desde un paquete se hace con el parámetro `-U`. En este caso, todos los ficheros se actualizan con los de la nueva versión: se suprimen los antiguos y los nuevos los sustituyen. Se guardan los antiguos ficheros de configuración con la extensión `.rpmsave`. Si el paquete no está instalado, la actualización asume el papel de instalación. Cuidado con este parámetro: instala el paquete incluso si la versión que actualiza no está instalada.

```
rpm -Uvh php-4.1.3-1.i586.rpm
```

La actualización es también posible con `-F`. Pero si el paquete no está instalado, tampoco se instalará durante la actualización, al contrario que con `-U`. Así, si dispone de todos los paquetes de actualización del sistema y usted sólo desea actualizar los que realmente están instalados, entonces puede teclear:

```
rpm -Fvh *.rpm
```

La supresión se efectúa con el parámetro `-e`. Sin embargo, cuidado: debe pasar como argumento el nombre del paquete, y no el nombre del fichero de paquete.

```
rpm -e php
```

Cuenta con varias opciones:

- `--force`: en caso de conflicto con otro paquete (el caso más habitual es el de dos paquetes que proponen el mismo fichero en el mismo sitio), esta opción fuerza la instalación.
- `--nodeps`: si el paquete rechaza instalarse por culpa de un problema de dependencias, esta opción forzará la instalación. Ocurre a veces que este error se produce cuando se instala la dependencia en cuestión de otra manera que no es a partir de un paquete rpm (p. ej.: compilación, binario copiado a mano).

---

 En la medida de lo posible, evite utilizar estas opciones, que pueden romper algunas dependencias, en particular si utiliza un sistema de metapaquete (yum, apt, zypper, urpmi, etc.).

---

### 4. El caso del núcleo

La instalación o la actualización de un núcleo es un caso particular. En efecto, la actualización suprime la versión antigua. El núcleo es un componente muy crítico del sistema. Si, como resultado de la actualización, el sistema no funciona (o funciona mal), habría que volver a instalar un antiguo núcleo desde el soporte de

instalación. El procedimiento es el siguiente:

- Instalación del nuevo núcleo con el parámetro `-i`, se añadirá al sistema.
- Reinicio y prueba de los programas y periféricos con el nuevo núcleo.
- Si funciona correctamente, supresión eventual del antiguo núcleo con `-e`.
- Edición de `/boot/grub/grub.conf` y modificación de la línea **Default** para arrancar por defecto en el nuevo núcleo.

## 5. Consultas RPM

Se puede consultar la base de datos RPM con el parámetro `-q` seguido de algunas opciones.

`-a`: lista de todos los paquetes instalados.

`-i`: información general (el resumen) del paquete.

`-l`: lista de los ficheros instalados.

`-f nombre`: encuentra el paquete que contiene el fichero dado.

`-p nombre`: la búsqueda se efectúa en el fichero del paquete dado.

`--requires`: dependencias del paquete.

`--provides`: lo que proporciona el paquete.

`--scripts`: scripts ejecutados en la instalación y la supresión.

`--changelog`: el historial del paquete.

```
$ rpm -qilp libjpeg-6.2.0-738.i586.rpm
Name        : libjpeg                                Relocations: (not relocateable)
Version     : 6.2.0                                  Vendor: SUSE LINUX Products
GmbH, Nuernberg, Germany
Release     : 738                                    Build Date: Sat Mar 19 20:07:55
2005
Install date: (not installed)                      Build Host: d121.suse.de
Group       : System/Libraries                      Source RPM: jpeg-6b-738.src.rpm
Size        : 125804                                 License: BSD, Other License(s),
see package
Signature   : DSA/SHA1, Sat Mar 19 20:12:06 2005, Key ID a84edae89c800aca
Packager    : http://www.suse.de/feedback
URL         : http://www.ijg.org/
Summary     : JPEG libraries
Description :
The libraries (static and dynamic) for the jpeg-graphics format. The
sources are contained in the jpeg source package.
```

Authors:

-----
Rob Hooft <hooft@EMBL-Heidelberg.DE>
Michael Mauldin <mlm@cs.cmu.edu>
/usr/lib/libjpeg.so.62
/usr/lib/libjpeg.so.62.0.0

```
$ rpm -qp libjpeg-6.2.0-738.i586.rpm -requires
/sbin/ldconfig
```

```

/sbin/ldconfig
rpmlib(PayloadFilesHavePrefix) <= 4.0-1
rpmlib(CompressedFileNames) <= 3.0.4-1
libc.so.6
libc.so.6(GLIBC_2.0)
libc.so.6(GLIBC_2.1.3)
rpmlib(PayloadIsBzip2) <= 3.0.5-1

$ rpm -qi php
Name        : php                                Relocations: (not relocateable)
Version     : 4.1.2                               Vendor: Red Hat, Inc.
Release     : 2.1.8                               Build Date: mer 14 jui 2004
11:24:16 CEST
Install date: lun 27 jun 2005 19:36:32 CEST      Build Host: porky.build.
redhat.com
Group       : Development/Languages             Source RPM: php-4.1.2-2.1.8.src.
rpm
Size        : 3843552                            License: The PHP License,
version 2.02
Packager    : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
URL         : http://www.php.net/
Summary     : The PHP HTML-embedded scripting language. (PHP: Hypertext
Preprocessor)
Description :
PHP is an HTML-embedded scripting language. PHP attempts to make it
easy for developers to write dynamically generated webpages. PHP also
offers built-in database integration for several commercial and
non-commercial database management systems, so writing a
database-enabled webpage with PHP is fairly simple. The most common
use of PHP coding is probably as a replacement for CGI scripts. The
mod_php module enables the Apache Web server to understand and process
the embedded PHP language in Web pages.

$ rpm -qa | grep php
php-ldap-4.1.2-2.1.8
php-imap-4.1.2-2.1.8
asp2php-0.75.17-1
php-4.1.2-2.1.8

$ rpm -qf /usr/bin/passwd
passwd-0.68-1.2.1

```

## 6. Verificación de los paquetes

Es posible que, después de la instalación de un paquete, se haya alterado uno o varios ficheros instalados (cambio de permisos, de propietario, edición, supresión, etc.). Como la base RPM contiene toda la información necesaria, se puede pedir una comprobación con el parámetro **-V**.

```
$ rpm -V php
S.5....T c /etc/php.ini
```

Un punto significa que una etapa de comprobación está OK. Si no es así:

- **s**: se modificó el tamaño del fichero.
- **5**: la suma MD5 ya no se corresponde.
- **T**: la fecha de modificación ya no es la misma.
- **u**: se modificó el propietario.
- **G**: se modificó el grupo.

- **I**: se modificó el vínculo simbólico.
- **M**: se modificaron los permisos o el tipo del fichero.
- **D**: se modificó el periférico (major/minor).

La **c** indica que se trata de un fichero de configuración.

A menudo el editor de la distribución firma los ficheros de paquetes RPM para garantizar su integridad. Se puede comprobar la integridad de un paquete con una clave pública GPG que se tiene que haber cargado ya en el sistema.

```
gpg --import RPM-GPG-KEY
rpm --import RPM-GPG-KEY
rpm --checksig libjpeg-6.2.0-738.i586.rpm
```

## 7. Las dependencias

Si usted utiliza las herramientas gráficas que proporciona su distribución, éstas intentarán resolver las dependencias en lugar de usted. El comando **rpm** sólo no lo hace por defecto. Hay herramientas complementarias «frontend» como **yast**, **apt** o **yum** que lo hacen en su lugar. La distribución Red Hat facilitaba hasta las versiones 4 (RHEL) una herramienta llamada **rpmdb-redhat** para instalar de manera automática las dependencias mediante **rpm**. Esto implica en particular el hecho de que todos los paquetes de la distribución deben encontrarse en el mismo lugar (en el mismo directorio) y que el sistema sólo funciona con los paquetes oficiales de Red Hat. Se utiliza el parámetro **--aid**.

```
$ rpm -ivh --aid libjpeg-6.2.0-738.i586.rpm
```

## 8. Actualizaciones automatizadas

Cada distribución facilita ahora una herramienta de actualización interactiva o automatizada. La openSUSE propone **YOU** (*Yast Online Update*), la Red Hat propone **up2date**. Al ser de pago la versión RHEL de Red Hat, el acceso a las actualizaciones depende de un número de licencia y de una inscripción a **RHN** (*Red Hat Network*). Las versiones derivadas, como CentOS, no se pueden actualizar mediante RHN, pero ofrecen su propio sitio remoto de actualización. Un gran número de distribuciones recientes utilizan hoy en día PackageKit, disponible en la siguiente dirección: <http://www.packagekit.org/>

# YUM

YUM es a los ficheros **rpm** lo que APT a los ficheros **dpkg**: un programa de gestión de paquetes. Recupera los paquetes dentro de los repositorios y gestiona las dependencias en lugar de usted. YUM significa *Yellow dog Updater Modified*. Se utiliza principalmente en las distribuciones Redhat (versiones Enterprise) y Fedora, pero se puede utilizar en cualquier distribución de tipo RPM si los repositorios asociados lo soportan.

Los comandos y ejemplos siguientes se basan en un servidor Redhat Enterprise Linux 5. El fichero de configuración es **/etc/yum.conf**.

## 1. Configuración de los repositorios

Los repositorios se colocan o en el fichero de configuración principal, o en el directorio **/etc/yum.repos.d**. El formato es el siguiente:

```
[fedora]
name=Fedora $releasever - $basearch
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/
$releasever/Everything/$basearch/os/
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=fedora-
$releasever&arch=$basearch
enabled=1
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch
```

El repositorio se llama (nombre corto) **rhel5**.

- **name**: el nombre largo del repositorio, detallado.
- **baseurl**: la URL del repositorio.
- **gpgcheck**: requiere una verificación de la firma GPG del repositorio.
- **enabled**: si ausente o a 1, el repositorio está activo.
- **gpgkey**: ruta de la clave pública GPG.

Las URL de los repositorios pueden ser locales (`file://`) o remotas (`http://` o `ftp://`). Deben apuntar a un directorio que contiene la información de repositorios que se encuentran en la carpeta `repodata`.

El archivo de configuración puede hacer uso de variables, como en este caso `$basearch` (arquitectura de 32 o 64 bits) o `$releasever` (la versión de la distribución). El comando yum rellena automáticamente estas variables según la instalación. Por supuesto, se les puede asignar los valores que usted crea oportunos.

Teniendo en cuenta los valores por defecto, se puede declarar un simple repositorio de la manera siguiente:

```
[updates-rhel5]
name=UPDATES-RHEL5
baseurl=ftp://ftp.server.com/RPMS.rhel5_updates_x86
```

Cuidado, sin embargo, ya que la configuración de YUM puede modificar los valores por defecto. La sección `[main]` de **/etc/yum.conf** puede, así, contener la línea:

```
gpgcheck=1
```

En este caso tendrá que modificar el valor `gpgcheck` a 0 en los repositorios que no necesitan firma.

## 2. Utilización de los repositorios

### a. Refrescar la caché

Con cada comando, YUM intenta refrescar sus datos si se ha superado el plazo de expiración. Se puede reducir o ampliar este plazo con la modificación de la línea `metadata_expire` del fichero de configuración.

```
metadata_expire=1h
```

Puede forzar la actualización de la caché con el parámetro `makecache`:

```
root@slyserver ~]# yum makecache
Updates-rhel5      | 951 B    00:00
other.xml.gz       | 94 kB    00:00
rhel5              | 1.3 kB   00:00
other.xml.gz       | 5.9 MB   00:00
rhel5-VT           | 1.3 kB   00:00
other.xml.gz       | 32 kB    00:00
updates-rhel5          14/14
rhel5                2255/2255
rhel5-VT             35/35
Metadata Cache Created
```

Otra posibilidad consiste en forzar la supresión de la caché para que se vuelva a construir de manera automática con el próximo comando YUM:

```
root@slyserver ~]# yum clean all
Cleaning up Everything
```

### b. Listar los paquetes

El parámetro `list` permite listar los paquetes. Todos se listan por defecto. Usted puede precisar una lista de paquetes o facilitar caracteres comodines. Hay varias opciones disponibles:

- **all**: es el caso por defecto: se listan primero los paquetes instalados y luego los disponibles para la instalación.
- **available**: los paquetes disponibles para instalación.
- **updates**: los paquetes que se pueden actualizar.
- **installed**: los paquetes actualizados.
- **obsoletes**: los paquetes del sistema ahora obsoletos debido a las versiones superiores disponibles.
- **recent**: los últimos paquetes añadidos en los repositorios.
- **repolist**: los repositorios configurados en `yum.conf` o `yum.repos.d`.

El comando siguiente lista los núcleos disponibles:

```
[root@slyserver etc]# yum list available kernel\*
Available Packages
kernel-PAE.i686          2.6.18-128.el5        rhel5
```

El

kernel-PAE-devel.i686	2.6.18-128.el5	rhel5
kernel-debug.i686	2.6.18-128.el5	rhel5
kernel-debug-devel.i686	2.6.18-128.el5	rhel5
kernel-devel.i686	2.6.18-128.el5	rhel5
kernel-doc.noarch	2.6.18-128.el5	rhel5
kernel-xen.i686	2.6.18-128.el5	rhel5
kernel-xen-devel.i686	2.6.18-128.el5	rhel5

parámetro `info` devuelve la información detallada de un paquete. Es equivalente al parámetro `-i` del comando `rpm`. Así, para el paquete `mc`, obtendrá algo parecido a lo siguiente:

```
[root@slyserver ~]# yum info mc
Installed Packages
Name        : mc
Arch        : i386
Epoch       : 1
Version     : 4.6.1a
Release     : 35.el5
Size        : 5.2 M
Repo        : installed
Summary     : Shell visual y gestor de ficheros en consola
texto ergonómico
URL         : http://www.ibiblio.org/mc/
License     : GPL
Description: Midnight Commander es una shell visual comparable a un
gestor de ficheros salvo que ofrece muchas más
funciones. Se trata de una aplicación en modo carácter pero
integra también un soporte para ratón. Las funciones más
agradables de Midnight Commander son sus capacidades para efectuar
traspasos FTP, visualizar ficheros tar y zip, buscar
ficheros específicos en los RPM.
```

## c. Instalar paquetes

Pase el parámetro `install`, seguido de los nombres de paquetes a instalar.

A continuación tenemos un ejemplo de instalación del paquete `mc`:

```
[root@slyserver /etc]# yum install mc
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
--> Running transaction check
---> Package mc.i386 1:4.6.1a-35.el5 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package   Arch   Version      Repository  Size
=====
Installing:
mc        i386   1:4.6.1a-35.el5    rhel5      2.1 M

Transaction Summary
=====
Install      1 Package(s)
Update      0 Package(s)
Remove      0 Package(s)

Total download size: 2.1 M
Is this ok [y/N]: y
```

```
Downloading Packages:
mc-4.6.1a-35.el5.i386.rpm           2.1 MB      00:00
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : mc                      [1/1]

Installed: mc.i386 1:4.6.1a-35.el5
Complete!
```

## d. Actualizaciones

Compruebe la publicación de actualizaciones con el parámetro `check-update`:

```
[root@slyserver /etc]# yum check-update
```

Si no devuelve nada quiere decir que no hay publicada ninguna actualización.

Tiene dos posibilidades para instalar las actualizaciones:

- **update**: actualización de un paquete o de todos si no se especifica ningún paquete.
- **upgrade**: actualización completa de la distribución: se sustituyen los paquetes considerados como obsoletos con los de la última versión disponible.

En algunos casos, para el núcleo por ejemplo, tendrá que evitar durante una actualización instalar automáticamente algunos paquetes. En este caso, utilice el parámetro `--exclude`:

```
[root@slyserver etc]# yum list --exclude=kernel\* update
```

Para convertir esta exclusión en algo permanente, póngala en el fichero de configuración añadiendo una línea como la siguiente:

```
exclude=php* kernel*
```

## e. Buscar un paquete

Utilice el parámetro `search`, seguido del paquete o de los paquetes a buscar en los repositorios. Se autorizan los caracteres comodines.

```
root@slyserver etc]# yum search tomcat
=====
Matched: tomcat
jakarta-commons-collections-tomcat5.i386 : Jakarta Commons Collection dependency for Tomcat5
struts-webapps-tomcat5.i386 : Ejemplos de aplicaciones Web struts para tomcat5
tomcat5.i386 : Motor Servlet/JSP Apache, RI para Servlet 2.4/JSP 2.0 API
tomcat5-admin-webapps.i386 : Aplicaciones Web de administración para tomcat
tomcat5-common-lib.i386 : Librería necesaria durante la ejecución del container Tomcat Web
tomcat5-jasper.i386: Compilador JARs y scripts asociados para tomcat5
tomcat5-jasper-javadoc.i386: Documentación Javadoc para tomcat5-jasper
tomcat5-jsp-2.0-api.i386: Implementaciones de clases Jakarta Tomcat Servlet y JSP
tomcat5-jsp-2.0-api-javadoc.i386: Documentación Javadoc generada para tomcat5-jsp-2.0-api
tomcat5-server-lib.i386: Librería necesaria durante la ejecución del container Tomcat Web
tomcat5-servlet-2.4-api.i386: Implementación de las clases Servlets para Jakarta Tomcat
```

tomcat5-servlet-2.4-api-javadoc.i386: Documentación Javadoc  
generada para tomcat5-servlet-2.4-api  
tomcat5-webapps.i386: Aplicaciones Web para Jakarta Tomcat

## f. Suprimir un paquete

Para suprimir un paquete, utilice el parámetro `remove`:

```
[root@slyserver~]# yum remove mc
Setting up Remove Process
Resolving Dependencies
--> Running transaction check
---> Package mc.i386 1:4.6.1a-35.el5 set to be erased
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package      Arch      Version      Repository      Size
=====
Removing:
mc          i386      1:4.6.1a-35.el5   installed      5.2 M

Transaction Summary
=====
Install      0 Package(s)
Update      0 Package(s)
Remove      1 Package(s)

Is this ok [y/N]: y
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
Erasing       : mc                               [1/1]

Removed: mc.i386 1:4.6.1a-35.el5
Complete!
```

# Debian Package

## 1. dpkg: el gestor de paquetes Debian

El comando **dpkg** es el equivalente de rpm para las distribuciones Debian y derivados, entre las cuales se encuentra Ubuntu. Hace lo mismo o casi que **rpm**. Los paquetes Debian llevan una extensión .deb para reconocerlos y disponen de la misma información y mismos medios que un paquete **rpm**. El comando **dpkg** es el encargado de la instalación, la creación, la supresión y la gestión de los paquetes Debian.

Se suele colocar la base de datos **dpkg** en **/var/lib/dpkg**. Los ficheros de este directorio tienen formato de texto. Sin embargo, no edite los ficheros manualmente. El fichero **/var/lib/dpkg/status** contiene la totalidad de los paquetes conocidos por **dpkg** con su estado.

```
# grep ^Package: /var/lib/dpkg/status | grep glibc
Package: devhelp-book-glibc
Package: libg++2.8.1.3-glibc2.2
Package: glibc-doc
Package: libstdc++2.10-glibc2.2
```

Dpkg dispone de una interfaz gráfica, GDebi, que permite evitar la utilización de la línea de comandos.

## 2. Instalación, actualización y supresión

La opción **-i**, o **--install**, instala el paquete o los paquetes que se pasan como argumento.

```
# dpkg -i miapquete.deb
```

Observe que, al igual que rpm, **dpkg** no gestiona las dependencias. Si faltan dependencias, el comando le informará. En este caso, tendrá que instalar las dependencias de la misma manera antes de instalar su paquete.

Puede pedir la instalación de todos los paquetes presentes en el árbol con el parámetro **-R**, la inicial de Recursividad. En este caso, indique como argumento un nombre de directorio y se instalarán todos los paquetes presentes en el directorio, así como sus subdirectorios.

```
# dpkg -R directorio
```

Se actualiza de la misma manera que se instala, con **-i**. Si instala un paquete ya presente, **dpkg** efectúa su actualización. Así, una instalación o una actualización respeta la metodología siguiente:

- Extracción de los ficheros de control del nuevo paquete.
- Cuando ya se instaló una antigua versión del mismo paquete, se ejecuta el script presupresión del antiguo paquete.
- Ejecución del script de preinstalación si se facilita con el paquete.
- Desempaquetado de nuevos ficheros y salvaguardia de los antiguos para poder restaurarlos en caso de problema.
- Cuando ya se ha instalado una antigua versión del paquete, se ejecuta el script postsupresión del antiguo paquete.
- Configuración del paquete.

No existe equivalencia con el modo **freshen** (-F) de rpm. Si desea actualizar un paquete únicamente en caso

de que ya esté instalado, debe comprobar primero si lo está. Si es el caso, el paquete comienza por `ii` en la lista:

```
# (dpkg -l zip | grep ^ii >/dev/null) && echo PRESENT || echo ABSENT
PRESENT
# (dpkg -l slapd | grep ^ii >/dev/null) && echo PRESENT || echo ABSENT
ABSENT
```

Para actualizar un paquete sólo en el caso de que ya esté presente, utilice este tipo de línea de comandos o un equivalente de su invención:

```
# (dpkg -l zip | grep ^ii >/dev/null) && dpkg -l zip.deb
```

 Observe que lo más fácil para actualizar sus paquetes ya presentes es crear un repositorio APT que contenga sus actualizaciones y ejecutar un `apt-get upgrade` desde el cliente.

La supresión de un paquete se lleva a cabo con el parámetro `-r` (en minúscula). Esta vez, también le corresponde a usted gestionar las dependencias.

La supresión de un paquete sigue las etapas siguientes:

- Ejecución del script de presupresión.
- Supresión de los ficheros instalados.
- Ejecución del script de postsupresión.

```
# dpkg -r zip
```

Se suprime todo salvo los ficheros de configuración para evitar una nueva configuración de la herramienta si vuelve a instalarla. Para suprimir todo, incluyendo estos ficheros, precise el parámetro `-P` (purga).

```
# dpkg -P apache
```

 Cuidado: no confunda los parámetros `-r` y `-R`!

Si sustituye el nombre del paquete por los parámetros `-a` o `--pending`, se borran los paquetes no instalados (no desempaquetados) pero que están marcados en la base de datos para purgarlos o suprimirlos.

La utilización de las opciones `--force-all` y `--purge` permite forzar la desinstalación del paquete y suprimir los ficheros de configuración asociados.

```
# dpkg --force-all --purge nombre_del_paquete
```

### 3. Ejemplos con dpkg

#### a. Listar los paquetes

Puede listar todos los paquetes Debian conocidos del sistema con el parámetro `-l`:

```
# dpkg -l
...
ii  adduser          3.102
```

```
Add and remove users and groups          8.64
ii  alien                               8.64
install non-native packages with dpkg
rc  amavisd-new                         2.4.2-6.1
Interface between MTA and virus scanner/cont
ii  antlr                                2.7.6-7
language tool for constructing recognizers,
rc  apache                             1.3.34-4.1
versatile, high-performance HTTP server
ii  apache-common                      1.3.34-4.1+etch1
support files for all Apache web servers
ii  apache2                            2.2.3-4+etch4
Next generation, scalable, extendable web server
rc  apache2-common                     2.0.54-5sarge2
next generation, scalable, extendable web server
ii  apache2-mpm-prefork                2.2.3-4+etch4
Traditional model for Apache HTTPD 2.1
ii  apache2-utils                      2.2.3-4+etch4
utility programs for web servers
ii  apache2.2-common                  2.2.3-4+etch4
Next generation, scalable, extendable web server
...
...
```

Puede indicar un motivo particular:

```
# dpkg -l "apt*" |grep ^ii
ii  apt           0.6.46.4-0.1      Advanced front-end for dpkg
ii  apt-file      2.0.3-7        APT package searching utility -
command-line
ii  apt-listchanges 2.72.5etch2   Display change history from .deb
archives

ii  apt-rpm-repository 0.5.15lorg3.2-1 tools to create an APT RPM repository
ii  apt-utils      0.6.46.4-0.1      APT utility programs
ii  aptitude       0.4.4-4        terminal-based apt frontend
```

 Si su consola es demasiado pequeña para visualizar los nombres de los paquetes (segunda columna), puede valerse del truco siguiente:

```
# COLUMNS=160 dpkg -l "kernel*" | grep ^ii | awk '{print $2}'
kernel-image-2.6.7-1-686
kernel-image-2.6.8-1-686-smp
kernel-image-2.6.8-2-686
kernel-image-2.6.8-2-686-smp
```

Otro método consiste en emplear la opción **--get-selections**:

```
# dpkg --get-selections | grep kernel
fai-kernels                           install
kernel-image-2.6.7-1-686               install
kernel-image-2.6.8-1-686-smp           install
kernel-image-2.6.8-2-686               install
kernel-image-2.6.8-2-686-smp           install
linux-kernel-headers                  install
```

## b. Encontrar un paquete que contiene un fichero

El parámetro **-S** seguido del nombre de un fichero (su ruta) permite encontrar el paquete de origen.

```
# dpkg -S /usr/bin/basename  
coreutils: /usr/bin/basename
```

### c. Listar el contenido de un paquete

El parámetro `-L` lista el contenido del paquete o de los paquetes indicados:

```
# dpkg -L coreutils | grep bin  
/bin  
/bin/mkdir  
/bin/mv  
/bin/true  
/bin/mknod  
/bin/sleep  
/bin/touch  
/bin/chgrp  
/bin/uname  
/bin/echo  
/bin/sync  
/bin/ln  
/bin/date  
/bin/dir  
/bin/readlink  
...
```

## 4. Convertir paquetes

La herramienta **alien** permite convertir paquetes RPM en DPKG y viceversa. Algunos paquetes sólo se facilitan para un sistema u otro. Cuando se suministra un producto de software sólo bajo una forma, es molesto tener que instalarlo de todas maneras en otra plataforma Linux.

A continuación, ponemos el ejemplo de un paquete, el cliente Networker, proporcionado solamente para Red Hat. Con Alien, es posible convertirlo al formato dpkg.

El parámetro por defecto `-d` convierte del rpm al dpkg:

```
# alien -d lgtocnt-7.4-1.i686.rpm  
Warning: Skipping conversion of scripts in package lgtocnt: postinst postrm  
preinst prerun  
Warning: Use the --scripts parameter to include the scripts.  
lgtocnt_7.4-2_i386.deb generated
```

Como se ha indicado ya, la conversión por defecto va a comprobar las dependencias, pero no va a incluir los scripts de pre y postinstalación. Entonces debe precisar el parámetro `--scripts`.

```
# alien --scripts -d lgtocnt-7.4-1.i686.rpm  
lgtocnt_7.4-2_i386.deb generated  
# ls -l *.deb  
-rw-r--r-- 1 root root 29471546 2008-05-09 14:45 lgtocnt_7.4-2_i386.deb
```

resultado es el siguiente:

```
# dpkg -I lgtocnt_7.4-2_i386.deb  
nuevo paquete Debian, versión 2.0.  
tamaño 29471546 bytes: archivo de control = 4498 bytes.  
 923 bytes,   18 líneas    control  
 3142 bytes,   57 líneas    md5sums  
 4014 bytes,  148 líneas    * postinst          #!/bin/sh  
1362 bytes,   35 líneas    * postrm          #!/bin/sh
```

```
 317 bytes,    11 líneas * preinst          #!/bin/sh
1828 bytes,    52 líneas * prerm            #!/bin/sh
  61 bytes,     3 líneas   shlibs

Package: lgtocnt
Version: 7.4-2
Section: alien
Priority: extra
Arquitectura: i386
Depends: libc6 (>= 2.3.6-6), libgl1-mesa-glx | libgl1, libice6 (>=
1:1.0.0), libncurses5 (>= 5.4-5), libsm6, libx11-6, libxext6,
libxmu6, libxp6, libxrender1, libxt6
Installed-Size: 71632
Maintainer: root <root@s64p17bib76.dsit renfe.es>
Description: NetWorker Client
  EMC NetWorker protects the critical business data of more than 10,000
...
  and the smallest satellite branch offices.

(Converted from a rpm package by alien version 8.64.)
```

## 5. La herramienta dselect

La herramienta **dselect** es un frontend (como APT) para dpkg, que gestiona las dependencias y los conflictos. Históricamente fue la primera. Sin embargo, su sustituto APT tiene mejor calidad.

El manual de dselect indica de manera clara que hoy en día la herramienta ha caído en desuso.

- La interfaz es confusa.
- Hay poco mantenimiento.
- El manual es incompleto.
- No hay filtro.
- Los accesos ya no son estándares.

Por todos estos motivos es preferible utilizar APT.

Terminal

```
Archivo Editar Ver Terminal Solapas Ayuda
dselect - lista principal de paquetes (disp., prioridad)      marca:+/- literal:v ayuda:?
EIOM Pri Sección Paquete     Ver.inst.  Ver.disp.  Descripción
- Todos los paquetes -
--- Paquetes actualizados, no hay versión más nueva ---
--- Paquetes actualizados requeridos ---
    --- Paquetes actualizados requeridos en la sección admin ---
*** Req admin base-files 4        4      Debian base system miscellaneous files
*** Req admin dpkg      1.13.26   1.13.26  package maintenance system for Debian
*** Req admin dselect   1.13.26   1.13.26  user tool to manage Debian packages
*** Req admin e2fsprogs 1.39+1.40-W 1.39+1.40-W ext2 file system utilities and libraries
*** Req admin initscripts 2.86.ds1-38 2.86.ds1-38 Scripts for initializing and shutting down the system
*** Req admin libpam-runtime 0.79-5+etch 0.79-5+etch Runtime support for the PAM library
*** Req admin login     4.0.18.1-7+ 4.0.18.1-7+ system login tools
*** Req admin makedev   2.3.1-83   2.3.1-83  creates device files in /dev
*** Req admin mount    2.12r-19etc 2.12r-19etc Tools for mounting and manipulating filesystems
*** Req admin passwd   4.0.18.1-7+ 4.0.18.1-7+ change and administer password and group data
base-files instalado ; instalar (era: instalar). requeridos
base-files - Debian base system miscellaneous files

This package contains the basic filesystem hierarchy of a Debian system, and several important miscellaneous files, such as /etc/debian_version, /etc/host.conf, /etc/issue, /etc/motd, /etc/profile, /etc/nsswitch.conf, and others, and the text of several common licenses in use on Debian systems.
```

descripción de base-files

Se recomienda no seguir utilizando desect.

# Gestor APT

## 1. Fundamentos

Ya sea con rpm o dpkg, el problema sigue siendo el mismo: estas dos herramientas detectan las dependencias de los paquetes para autorizar o no su instalación, pero no las resuelven. Dicho de otro modo, si una dependencia de un paquete no está presente, no se instalará salvo si se resuelve la dependencia:

- o instalando previamente los paquetes que faltan,
- o indicando en la misma línea la ruta de estos mismos paquetes.

Asimismo, en el momento de la actualización se presenta un problema con los ficheros de configuración. ¿Qué se debe hacer?

APT permite resolver estos problemas gestionando las dependencias en lugar de usted. **APT** significa *Advanced Packaging Tool*. En vez de especificar un paquete (local o remoto), se encarga de los repositorios de paquetes situados en un CD, un DVD, en un directorio local, en una fuente remota en Internet (ftp, http), etc.

Un repositorio contiene un conjunto de paquetes que dependen o bien los unos de los otros, o bien de otros paquetes procedentes de otros repositorios. APT puede gestionar varios repositorios en varios sitios. Se las arregla solo: cuando usted instala un paquete, también instala sus dependencias (si las encuentra).

## 2. Los repositorios

### a. Configuración

Los repositorios quedan fijados en el fichero `/etc/apt/sources.list`. El siguiente fichero proviene de una instalación Debian Etch en el cual se añadieron los repositorios contrib y non-free.

```
$ cat /etc/apt/sources.list
## squeeze
deb http://ftp.es.debian.org/debian/ squeeze main contrib non-free
deb-src http://ftp.es.debian.org/debian/ squeeze main contrib non-free

# security
deb http://security.debian.org/ squeeze/updates main contrib non-free
deb-src http://security.debian.org/ squeeze/updates main contrib non-free
```

Los repositorios de software se ubican físicamente en el árbol de directorios del servidor. El comando **genbasedir** permite crear un repositorio. La sintaxis de una línea del fichero `sources.list` es la siguiente:

```
deb uri distribución componente1 componente2 ...
```

- `uri` es la ruta hacia la raíz del repositorio o de los repositorios. Puede ser una URL de tipo http o ftp, pero también una ruta local (file), un CD-Rom o DVD-Rom (CDrom), una ruta ssh, etc.
- La distribución es, como su nombre indica, el nombre de la distribución Debian. Aquí es **squeeze**, pero es posible especificar otras versiones de la distribución (testing, sarge, etc.) para poder recuperar paquetes de otros repositorios, más recientes por ejemplo. Se puede precisar la arquitectura. Si no es el caso, APT se las arregla solo para añadir el sufijo necesario.
- Los componentes son los nombres de los repositorios para la distribución dada.

En la práctica, la uri, la distribución y los componentes permiten reconstituir la URL completa de acceso al repositorio.

- Diríjase a <http://ftp.es.debian.org/debian/>.
- Haga clic en el archivo llamado **dists**. Contiene la lista de las distribuciones Debian.
- En **dists**, haga clic en **squeeze**, el nombre de la distribución actual.
- En **squeeze**, encontrará las carpetas **contrib**, **main** y **non-free**.

La línea `deb http://ftp.es.debian.org/debian/ squeeze main contrib non-free` corresponde por lo tanto a las URL:

- <http://ftp.es.debian.org/debian/dists/squeeze/main>
- <http://ftp.es.debian.org/debian/dists/squeeze/contrib>
- <http://ftp.es.debian.org/debian/dists/squeeze/non-free>

Si usted prosigue, por ejemplo entrando en **main**, encontrará una serie de carpetas con sufijos según la arquitectura de su instalación de Linux. La máquina VMWare de prueba es de tipo i386. Por lo tanto, se buscarán los paquetes binarios en **binary-i386**.

No se sorprenda de no encontrar paquetes en este último directorio, sino ficheros:

- **Release**: descripción del repositorio.
- **Packages.gz**: índice de los paquetes del repositorio en el formato gzip.
- **Packages.bz2**: lo mismo en el formato bzip2.

¿Dónde están realmente los paquetes? La respuesta está en los ficheros **Packages.\***. Las primeras líneas de uno de estos ficheros tienen este aspecto:

```
Package: 3270-common
Priority: optional
Section: net
Installed-Size: 96
Maintainer: Bastian Blank <waldi@debian.org>
Architecture: i386
Source: ibm-3270
Version: 3.3.4p6-3.3
Depends: libc6 (>= 2.3.6-6)
Recommends: x3270 (= 3.3.4p6-3.3) | c3270 (= 3.3.4p6-3.3), pr3287 (= 3.3.4p6-3.3)
Filename: pool/main/i/ibm-3270/3270-common_3.3.4p6-3.3_i386.deb
Size: 21910
MD5sum: 209bb0595c53421c433f4524147d6335
SHA1: c89e5ef06fa0978b5a0935c90273b5c5997b2142
SHA256: 881cf62382b9e1945155bdd366645d9660c1848aaab3a58e73d2bdcaa49301ae
Description: Common files for IBM 3270 emulators and pr3287
3270-common contains files referenced in other 3270 packages
```

La línea **Filename** le indica que el fichero está en `pool/main/i/ibm-3270/` desde el uri, y por lo tanto accesible desde la URL <http://ftp.es.debian.org/debian/pool/main/i/ibm-3270/>. Observe también la descripción de los paquetes, las líneas **Depends** y **Recommends** que permiten a APT resolver las dependencias. Este fichero se parece mucho, y con motivos, al fichero **status** de la base dpkg local.

## b. Actualización de la base de datos

Una vez configurados sus repositorios, debe actualizar la base de datos local de APT con el comando **apt-get** y la opción **update**.

```
root@debian:~# apt-get update
Des:1 http://ftp.es.debian.org squeeze Release.gpg [1672 B]
Ign http://ftp.es.debian.org/debian/ squeeze/main Translation-en
Des:2 http://ftp.es.debian.org/debian/ squeeze/main Translation-es [472 kB]
Des:3 http://security.debian.org squeeze/updates Release.gpg [836 B]
Ign http://security.debian.org/ squeeze/updates/main Translation-en
Ign http://security.debian.org/ squeeze/updates/main Translation-es
Des:4 http://security.debian.org squeeze/updates Release [86,9 kB]
Des:5 http://security.debian.org squeeze/updates/main Sources [57,8 kB]
Des:6 http://security.debian.org squeeze/updates/main amd64 Packages [177 kB]
Des:7 http://ftp.es.debian.org squeeze-updates Release.gpg [836 B]
Ign http://ftp.es.debian.org/debian/ squeeze-updates/main Translation-en
Ign http://ftp.es.debian.org/debian/ squeeze-updates/main Translation-es
Des:8 http://ftp.es.debian.org squeeze Release [87,3 kB]
Des:9 http://ftp.es.debian.org squeeze-updates Release [113 kB]
Des:10 http://ftp.es.debian.org squeeze/main Sources [4540 kB]
Des:11 http://ftp.es.debian.org squeeze/main amd64 Packages [6540 kB]
Des:12 http://ftp.es.debian.org squeeze-updates/main Sources/DiffIndex [2023 B]
Des:13 http://ftp.es.debian.org squeeze-updates/main amd64 Packages/DiffIndex [2023 B]
Des:14 http://ftp.es.debian.org squeeze-updates/main amd64 Packages [9370 B]
Descargados 12,1 MB en 22seg. (538 kB/s)
Leyendo lista de paquetes... Hecho
root@debian:~#
```

### 3. Actualización de la distribución

Una vez actualizados los repositorios, puede actualizar en un solo comando todos los paquetes instalados en su distribución: APT verifica si unos paquetes más recientes están disponibles en los repositorios. Se basa para ello en la base de datos local. Si no está actualizada, es posible que ya no se encuentren algunos de los paquetes más antiguos.

Ejecute el comando **apt-get** con la opción **upgrade**. APT le informa que se puede actualizar ocho paquetes. Puede aceptar o rechazar. Si acepta, APT descarga estos paquetes y sus posibles dependencias, y los instala. El proceso puede ser más o menos largo según el número de actualizaciones y el tipo de soporte.

```
# apt-get upgrade
Leyendo lista de paquetes... Hecho
Creando Árbol de dependencias
Leyendo la información de estado... Hecho
Se actualizarán los siguientes paquetes:
  libjasper-runtime libjasper1 libsoup-gnome2.4-1 libsoup2.4-1 x11-common
  xbase-clients xorg xserver-xorg xserver-xorg-input-all
  xserver-xorg-video-all
10 actualizados, 0 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 867 kB de archivos.
Se utilizarán 131 kB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? S
Des:1 http://security.debian.org/ squeeze/updates/main libjasper1
amd64 1.900.1-7+squeezel [157 kB]
Des:2 http://security.debian.org/ squeeze/updates/main libjasper-runtime
amd64 1.900.1-7+squeezel [26,2 kB]
Des:3 http://security.debian.org/ squeeze/updates/main libsoup2.4-1
amd64 2.30.2-1+squeezel [176 kB]
Des:4 http://security.debian.org/ squeeze/updates/main libsoup-gnome2.4-1
amd64 2.30.2-1+squeezel [41,1 kB]
Des:5 http://security.debian.org/ squeeze/updates/main x11-common all
1:7.5+8+squeezel [279 kB]
Des:6 http://security.debian.org/ squeeze/updates/main xbase-clients all
1:7.5+8+squeezel [38,2 kB]
```

```

Des:7 http://security.debian.org/ squeeze/updates/main xserver-xorg-video-all
amd64 1:7.5+8+squeezel [32,0 kB]
Des:8 http://security.debian.org/ squeeze/updates/main xserver-xorg-input-all
amd64 1:7.5+8+squeezel [31,8 kB]
Des:9 http://security.debian.org/ squeeze/updates/main xserver-xorg
amd64 1:7.5+8+squeezel [53,3 kB]
Des:10 http://security.debian.org/ squeeze/updates/main xorg
amd64 1:7.5+8+squeezel [32,5 kB]
Descargados 867 kB en 4seg. (201 kB/s)
Leyendo lista de cambios... Hecho.
Preconfigurando paquetes ...
(Leyendo la base de datos ... 195159 ficheros o directorios instalados actualmente.)
Preparando para reemplazar libjasper1 1.900.1-7+b1
(usingo .../libjasper1_1.900.1-7+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de libjasper1 ...
Preparando para reemplazar libjasper-runtime 1.900.1-7+b1
(usingo .../libjasper-runtime_1.900.1-7+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de libjasper-runtime ...
Preparando para reemplazar libsoup2.4-1 2.30.2-1
(usingo .../libsoup2.4-1_2.30.2-1+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de libsoup2.4-1 ...
Preparando para reemplazar libsoup-gnome2.4-1 2.30.2-1
(usingo .../libsoup-gnome2.4-1_2.30.2-1+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de libsoup-gnome2.4-1 ...
Preparando para reemplazar x11-common 1:7.5+8
(usingo .../x11-common_1%3a7.5+8+squeezel_all.deb) ...
Desempaquetando el reemplazo de x11-common ...
Preparando para reemplazar xbase-clients 1:7.5+8
(usingo .../xbase-clients_1%3a7.5+8+squeezel_all.deb) ...
Desempaquetando el reemplazo de xbase-clients ...
Preparando para reemplazar xserver-xorg-video-all 1:7.5+8
(usingo .../xserver-xorg-video-all_1%3a7.5+8+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de xserver-xorg-video-all ...
Preparando para reemplazar xserver-xorg-input-all 1:7.5+8
(usingo .../xserver-xorg-input-all_1%3a7.5+8+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de xserver-xorg-input-all ...
Preparando para reemplazar xserver-xorg 1:7.5+8
(usingo .../xserver-xorg_1%3a7.5+8+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de xserver-xorg ...
Preparando para reemplazar xorg 1:7.5+8
(usingo .../xorg_1%3a7.5+8+squeezel_amd64.deb) ...
Desempaquetando el reemplazo de xorg ...
Procesando disparadores para man-db ...
Configurando libjasper1 (1.900.1-7+squeezel) ...
Configurando libjasper-runtime (1.900.1-7+squeezel) ...
Configurando libsoup2.4-1 (2.30.2-1+squeezel) ...
Configurando libsoup-gnome2.4-1 (2.30.2-1+squeezel) ...
Configurando x11-common (1:7.5+8+squeezel) ...
Configurando xbase-clients (1:7.5+8+squeezel) ...
Configurando xserver-xorg-video-all (1:7.5+8+squeezel) ...
Configurando xserver-xorg-input-all (1:7.5+8+squeezel) ...
Configurando xserver-xorg (1:7.5+8+squeezel) ...
Configurando xorg (1:7.5+8+squeezel) ...

```

Otra posibilidad consiste en realizar una actualización a fondo (llamada actualización remota). APT guarda una cierta coherencia en los paquetes durante la actualización, en particular en cuanto a la versión de la distribución. Usted puede especificar varias distribuciones Debian en sus repositorios. Pero incluso si una distribución es más reciente, un simple upgrade no va a transformar la suya en la última. Puede pedir a APT que fuerce la actualización hacia la nueva distribución con un **dist-upgrade**.

Para los objetivos de este libro, se han añadido los repositorios de la versión de prueba de Debian llamada testing:

Efectúe una actualización de la base. Observe que para ello ha hecho falta ampliar la caché de APT y limpiar la

```
## testing
deb http://ftp.es.debian.org/debian/ testing main contrib non-free
deb-src http://ftp.es.debian.org/debian/ testing main contrib non-free
# security testing
deb http://security.debian.org/ testing/updates main contrib non-free
deb-src http://security.debian.org/ testing/updates main contrib non-free
```

base:

```
# apt-get update
echo 'APT::Cache-Limit "141943904";' > /etc/apt/apt.conf.d/00Cache
# apt-get clean
# apt-get update
```

luego

```
$ apt-get dist-upgrade
Lectura de las listas de paquetes...
Construcción del arbol de las dependencias...
...
731 actualizados, 228 nuevamente instalados, 23 a quitar y 0 no actualizados.
Es necesario coger 858 Mb en las carpetas.
Después del desempaquetado, se utilizarán 623 Mb de espacio en disco adicional.
¿Desea proseguir [S/n]? S
...
```

Mucha suerte!

#### 4. Buscar e instalar un paquete individual

El comando **apt-cache** permite buscar un paquete por su nombre o su comentario en la base de datos local APT.

```
root@debian:~# apt-cache search torrent
aria2 - High speed download utility
vuze - Multimedia BitTorrent client
bitstormlite - BitTorrent Client based on C++/Gtk+2.0
bittornado-gui - bittorrent client with GUI interface
bittornado - bittorrent client (and tracker) with console and curses interfaces
bittorrent-gui - Original BitTorrent client - GUI tools
bittorrent - Original BitTorrent client - console tools
buildtorrent - command line torrent creation program
ctorrent - BitTorrent Client written in C++
debtorrent - bittorrent proxy for downloading Debian packages
deluge-common - bittorrent client written in Python/PyGTK (common files)
deluge-console - bittorrent client written in Python/PyGTK (console ui)
...
```

comando **apt-get install** xxx instala el paquete xxx:

```
root@debian:~# apt-get install vim-gtk
Leyendo lista de paquetes... Hecho
Creando Árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libruby1.8 vim-gui-common vim-runtime
Paquetes sugeridos:
  cscope vim-doc
```

```

Se instalarán los siguientes paquetes NUEVOS:
  libruby1.8 vim-gtk vim-gui-common vim-runtime
0 actualizados, 4 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 9570 kB de archivos.
Se utilizarán 36,6 MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]?
...

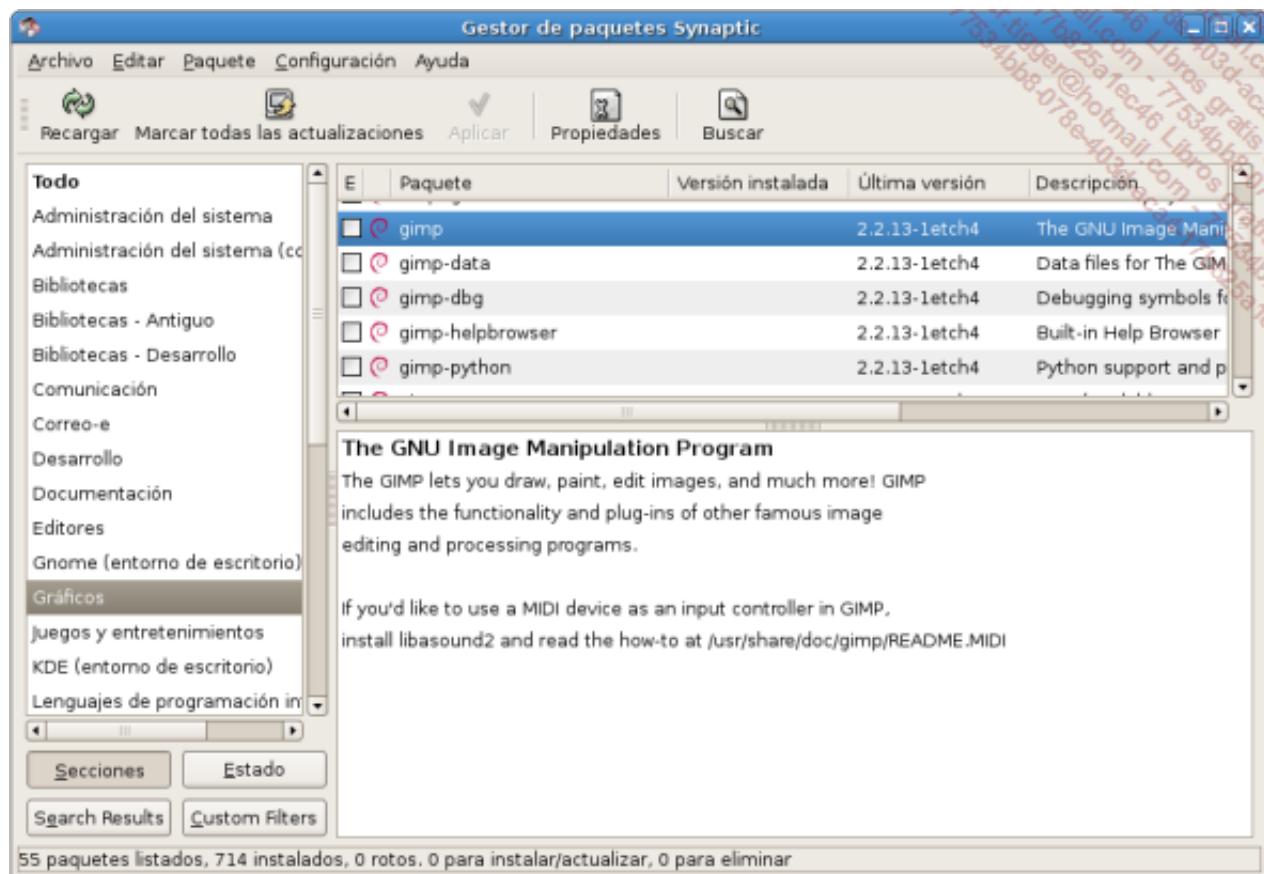
```

Dos opciones merecen recordarse:

- la **-s** para la simulación: APT indica lo que debería hacer, pero no lo hace.
- la **-f** para «fix-broken»: APT intenta arreglar los problemas de dependencias como puede (añadido de paquetes).

## 5. Cliente gráfico

La herramienta synaptic es un front-end: una interfaz gráfica que llama a las funciones de APT. Soporta todas las operaciones propuestas por APT y, además, disfruta de una interfaz muy atractiva.



Synaptic es un front-end para APT.

# Instalar desde las fuentes

## 1. Obtener las fuentes

A veces no es posible obtener un programa o una librería desde un paquete para su distribución. En este caso, queda la solución de compilar e instalar uno mismo el programa desde las fuentes.

Eso es posible para la mayoría de los programas bajo Linux gracias a las ventajas de los programas libres y de la licencia GPL tal como quedó definida en el primer capítulo. Cualquier programa libre va acompañado con sus fuentes. Por lo tanto, usted mismo puede volver a reconstruir el programa al compilarlo.

Es posible obtener un archivo fuente en diversos sitios de Internet, como por ejemplo SourceForge. Suele ser un archivo comprimido en formato tgz (archivo tar comprimido con gzip) o tar.bz2 (archivo tar comprimido en formato bzip2). Contiene:

- el código fuente en forma de ficheros .c, .h, .cpp, etc., según el lenguaje;
- a veces un fichero **Makefile** que permite automatizar la compilación del producto;
- a menudo un fichero **.configure** que permite generar el fichero Makefile en función de su instalación y de varias opciones.

## 2. Requisitos y dependencias

Para compilar su producto, tiene que respetar unos requisitos:

- presencia de la herramienta make;
- presencia del compilador o de los compiladores necesarios, en particular gcc;
- presencia de las dependencias: librerías, intérpretes, etc.

Este último punto es muy importante. Si falta una dependencia, se arriesga a encontrarse con varios problemas:

- no logrará preparar las fuentes para la compilación;
- la compilación generará errores;
- se compilará el programa pero con menos opciones;
- no se iniciará el binario resultante.

El comando **./configure** le proporcionará las dependencias que faltan y su versión si es posible. En este caso, usted puede o bien instalarlas desde los paquetes de su distribución, o bien instalarlas desde las fuentes.

---

 Cuando compila desde las fuentes sin pasar por su gestor de paquetes, pierde una parte de la gestión de las dependencias. Cuando instala paquetes que dependen de una versión de la herramienta instalada desde las fuentes, es posible, si las dependencias se basan en la existencia de un paquete y no de un fichero, que el gestor le impida instalar su paquete. Busque correctamente entre los repositorios, oficiales o no, si el programa existe en forma de paquete antes de volver a compilarlo o si existe un paquete fuente.

---

En todos los casos, no hace falta que sea root para compilar su programa. Sin embargo, según el destino de la aplicación, tendrá que pasar a root para finalizar la instalación.

### 3. Ejemplo de instalación

Va a compilar e instalar el producto PDFedit que permite editar y crear ficheros PDF.

- Descárguelo desde el vínculo siguiente. La versión probada es la versión 0.4.5:  
[http://sourceforge.net/project/showfiles.php?group\\_id=177354](http://sourceforge.net/project/showfiles.php?group_id=177354)

```
seb@debian:~$ ls -l pdfedit.tar.bz2
-rw-r--r-- 1 seb seb 21742 dic 17 17:52 pdfedit.tar.bz2
```

- Descomprima el fichero:

```
seb@debian:~/pdfedit$ rm -rf pdfedit-0.4.5
seb@debian:~/pdfedit$ tar xvjf pdfedit-0.4.5.tar.bz2
pdfedit-0.4.5/COPYING
pdfedit-0.4.5/Changelog
pdfedit-0.4.5/Makefile
pdfedit-0.4.5/Makefile.flags.in
pdfedit-0.4.5/Makefile.rules.in
pdfedit-0.4.5/README
pdfedit-0.4.5/config/
pdfedit-0.4.5/config/xpdf.m4
pdfedit-0.4.5/config/pdfedit-core-dev.m4
pdfedit-0.4.5/config/macro.m4
...
```

- Vaya a la carpeta **pdfedit-0.4.5** creada por la descompresión:

```
seb@debian:~/pdfedit$ cd pdfedit-0.4.5/
seb@debian:~/pdfedit/pdfedit-0.4.5$ ls -l
total 400
-rw-r--r-- 1 seb seb 8880 may 11 2010 Changelog
drwxr-xr-x 2 seb seb 4096 may 11 2010 config
-rwxr-xr-x 1 seb seb 284547 may 11 2010 configure
-rw-r--r-- 1 seb seb 20849 feb 23 2010 configure.in
-rw-r--r-- 1 seb seb 969 feb 21 2008 COPYING
drwxr-xr-x 5 seb seb 4096 may 11 2010 doc
-rwxr-xr-x 1 seb seb 4275 may 11 2010 getversion
-rw-r--r-- 1 seb seb 3451 nov 13 2008 Makefile
-rw-r--r-- 1 seb seb 9351 feb 23 2010 Makefile.flags.in
-rw-r--r-- 1 seb seb 1455 oct 21 2009 Makefile.rules.in
drwxr-xr-x 11 seb seb 4096 may 11 2010 projects
-rw-r--r-- 1 seb seb 21822 feb 23 2010 README
-rw-r--r-- 1 seb seb 15 may 11 2010 release_stamp
drwxr-xr-x 11 seb seb 4096 may 11 2010 src
drwxr-xr-x 3 seb seb 4096 may 11 2010 tools
-rw-r--r-- 1 seb seb 62 oct 30 2008 version_exclude
```

- Observe la presencia del fichero **configure** que es ejecutable.

```
seb@debian:~/pdfedit/pdfedit-0.4.5$ ./configure --help
`configure' configures PDFedit 0.4.5 to adapt to many kinds of systems.

Usage: ./configure [OPTION]... [VAR=VALUE]...

To assign environment variables (e.g., CC, CFLAGS...), specify them as
VAR=VALUE. See below for descriptions of some of the useful variables.

Defaults for the options are specified in brackets.

Configuration:
 -h, --help           display this help and exit
```

```

--help=short      display options specific to this package
--help=recursive display the short help of all the included
packages
-V, --version    display version information and exit
-q, --quiet, --silent do not print `checking...' messages
--cache-file=FILE cache test results in FILE [disabled]
-C, --config-cache alias for `--cache-file=config.cache'
-n, --no-create   do not create output files
--srcdir=DIR      find the sources in DIR
[configure dir or `...']

Installation directories:
--prefix=PREFIX    install architecture-independent files
in PREFIX
...

```

Una opción importante de configure es **--prefix**. Define la ubicación de la instalación una vez compilado el producto. Por defecto, el programa se instala en **/usr/local/**.

- Ejecute **./configure** solo. Le informará de las dependencias que faltan si es el caso.

```

root@debian:/home/seb/pdfedit/pdfedit-0.4.5# ./configure
checking for g++... g++
checking whether the C++ compiler works... yes
checking for C++ compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C++ compiler... yes
checking whether g++ accepts -g... yes
checking for gcc... gcc
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
checking for ranlib... ranlib
checking whether ln -s works... yes
checking how to run the C++ preprocessor... g++ -E
checking for grep that handles long lines and -e... /bin/grep
checking for egrep... /bin/grep -E
checking for ANSI C header files... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes
checking for unistd.h... yes
checking limits.h usability... yes
checking limits.h presence... yes
checking for limits.h... yes
checking for stdlib.h... (cached) yes
checking for string.h... (cached) yes
checking for unistd.h... (cached) yes
checking for stdbool.h that conforms to C99... yes
checking for _Bool... no
checking for an ANSI C-conforming const... yes
checking for inline... inline
checking for size_t... yes
checking whether struct tm is in sys/time.h or time.h... time.h
checking how to run the C preprocessor... gcc -E
checking if zlib is wanted... checking for inflateEnd in -lz... yes

```

```

checking zlib.h usability... yes
checking zlib.h presence... yes
checking for zlib.h... yes
checking for inflateEnd in -lz... (cached) yes
checking zlib in /usr... ok
checking for boostlib >= 1.20.0... configure: error: We could not
detect the boost libraries (version 1.20 or higher). If you have
a staged boost library (still not installed) please specify $BOOST_ROOT
in your environment and do not give a PATH to --with-boost option.
If you are sure you have boost installed, then check your version number
looking in <boost/version.hpp>. See http://randspringer.de/boost for more
documentation.

```

- Fíjese en el bloque en negrita: falta una librería. Tiene dos opciones:
  - instalar la librería que falta (y su paquete de desarrollo) bien desde las fuentes, bien desde el gestor de paquetes de su distribución;
  - no instalarlo: no es vital. Sin embargo, no habrá soporte de las fuentes Type1, lo que es un problema para los PDF.
- Despues de haber resuelto, si lo desea, las dependencias, debe volver a ejecutar el comando **./configure**. El comando **configure** crea el fichero **Makefile** correspondiente. Este fichero contiene el conjunto de las reglas, rutas y opciones para compilar el programa. El comando **make** depende de él.
- Inicie la compilación con el comando **make**. Puede que aparezcan alertas (líneas warning): no significa obligatoriamente que el programa no compilará o no funcionará posteriormente. De todas maneras, si la compilación produce errores, se parará por sí misma con un mensaje de error del compilador que puede a veces (pero no siempre) ponerle sobre la pista de una solución.

La compilación puede ser más o menos larga según el producto compilado. Para PDFEdit, una máquina Intel Core 2 Duo a 3200 MHz necesitó unos 6 minutos.

```

root@debian:/home/seb/pdfedit/pdfedit-0.4.5# make
cd /home/seb/pdfedit/pdfedit-0.4.5/src && make
make[1]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/src'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/xpdf && make staticlib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/src/xpdf'
cd goo && make
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/goo'
make[3]: No se hace nada para `all'.
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/goo'
cd fofi && make
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/fofi'
make[3]: No se hace nada para `all'.
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/fofi'
cd splash && make
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/splash'
make[3]: No se hace nada para `all'.
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/splash'
cd xpdf && make staticlib
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/xpdf'
make[3]: No se hace nada para `staticlib'.
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/xpdf'
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5

```

```
/src/xpdf'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/utils && make staticlib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/utils'
make[2]: No se hace nada para `staticlib'.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/utils'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/kernel && make staticlib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/kernel'
make[2]: No se hace nada para `staticlib'.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/kernel'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/xpdf && make extralib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf'
cd xpdf && make extralib
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/xpdf'
make[3]: No se hace nada para `extralib'.
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf/xpdf'
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/xpdf'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/tests/ && make
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/tests'
make[2]: No se hace nada para `all'.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/tests'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/kpdf-kde-3.3.2 && make -f
Makefile.gui staticlib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/kpdf-kde-3.3.2'
make[2]: No se hace nada para `staticlib'.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/kpdf-kde-3.3.2'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/qsa && make staticlib
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa'
make -f Makefile.qsa
make[3]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa'
cd src && make -f Makefile
make[4]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa/src'
cd qsa && make -f Makefile
make[5]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa/src/qsa'
make[5]: No se hace nada para `first'.
make[5]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa/src/qsa'
make[4]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa/src'
make[3]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa'
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/qsa'
cd /home/seb/pdfedit/pdfedit-0.4.5/src/gui && make -f Makefile.gui
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/gui'
make[2]: No se hace nada para `first'.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5
/src/gui'
make[1]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/src'
```

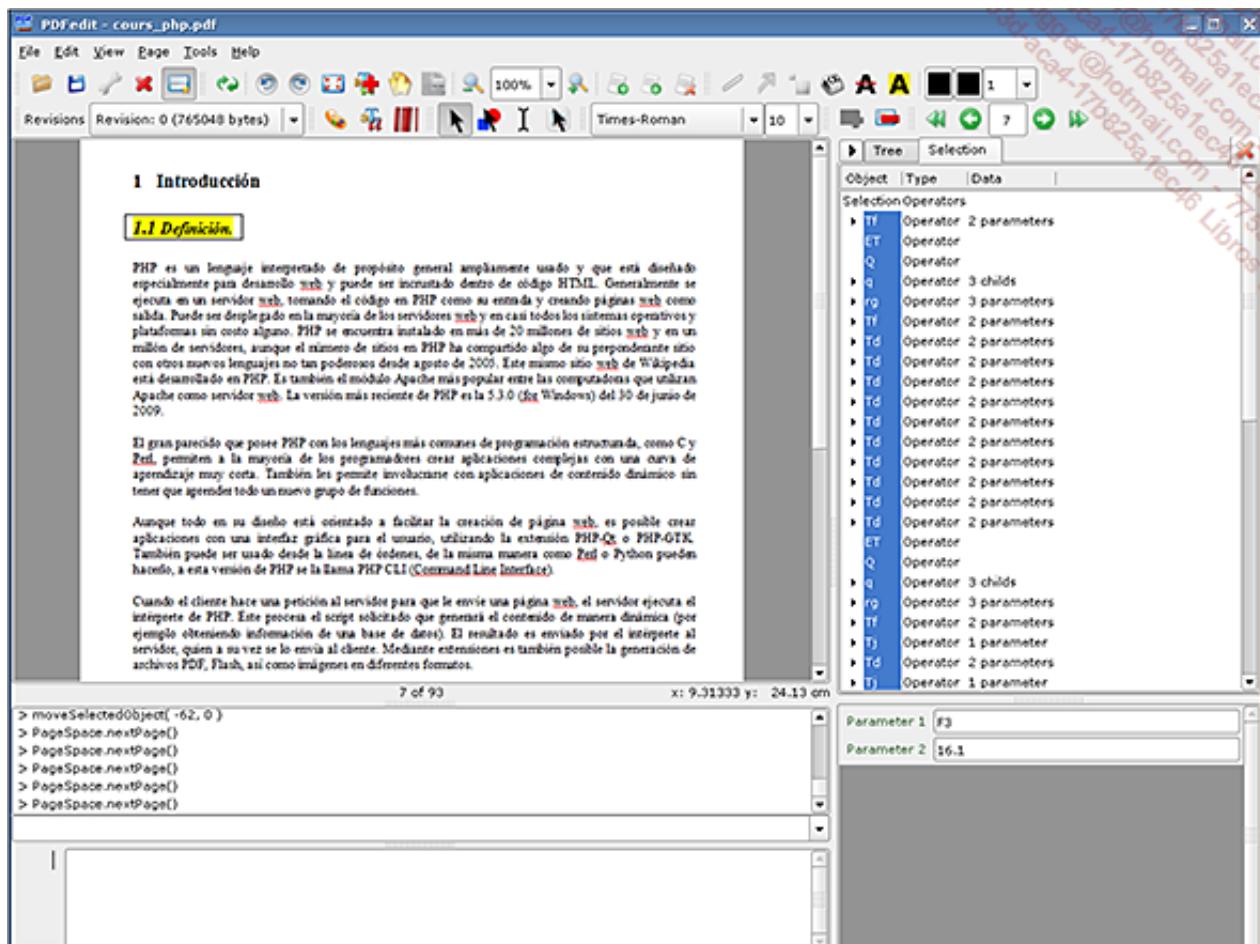
- Al finalizar sin error la compilación, termine la instalación del producto con **make install**. Cuidado: el producto se va a instalar en /usr/local/, lo que requiere los derechos del usuario root. Según su distribución, deberá usar o bien **su**, o bien el comando **sudo**.

```
seb@debian:~/pdfedit/pdfedit-0.4.5$ su -c "make install"
...
cp -f "lang/pdfedit_es.qm" "/usr/local/share/pdfedit/lang/"
cp -f "lang/pdfedit_fr.qm" "/usr/local/share/pdfedit/lang/"
cp -f "lang/pdfedit_ru.qm" "/usr/local/share/pdfedit/lang/"
cp -f "lang/pdfedit_sk.qm" "/usr/local/share/pdfedit/lang/"
cp -f "lang/pdfedit_zh_TW.qm" "/usr/local/share/pdfedit/lang/"
cp -f "scripts/delinearize.qs" "/usr/local/share/pdfedit/scripts/"
cp -f "scripts/flatten.qs" "/usr/local/share/pdfedit/scripts/"
cp -f "scripts/pdftoxml.qs" "/usr/local/share/pdfedit/scripts/"
cp -f "scripts/update.qs" "/usr/local/share/pdfedit/scripts/"
cp -f "pdfedit" "/usr/local/bin/pdfedit"
make[1]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/src/gui'
cd /home/seb/pdfedit/pdfedit-0.4.5/doc && make doc_dist
make[1]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc'
cd user && make pdfedit.1
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc/user'
make[2]: `pdfedit.1' está actualizado.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc/user'
make[1]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc'
cd /home/seb/pdfedit/pdfedit-0.4.5/doc && make doc_dist_install
make[1]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc'
cd user && make pdfedit.1
make[2]: se ingresa al directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc/user'
make[2]: `pdfedit.1' está actualizado.
make[2]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc/user'
mkdir -p /usr/local/share/man/man1;
cp -f user/pdfedit.1 /usr/local/share/man/man1
mkdir -p /usr/local/share/doc/pdfedit/;
cp -f LICENSE.GPL ../README AUTHORS README-* \
      README.cygwin /usr/local/share/doc/pdfedit
make[1]: se sale del directorio `/home/seb/pdfedit/pdfedit-0.4.5/doc'
seb@debian:~/pdfedit/pdfedit-0.4.5$
```

- Inicie el producto:

```
$ pdfedit
```

Felicidades, la compilación y la instalación han funcionado perfectamente.



*PDFEdit, compilado e instalado desde las fuentes*

## 4. Desinstalación

La mayoría de los **Makefile**, al menos los generados por **configure**, permiten la desinstalación. Se efectúa con el comando **make uninstall**.

```
$ su -c "make uninstall"
password:
...
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_commandline.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_editor.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_execute.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_lookandfeel.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_objecttree.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_paths.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/options_toolbars.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/pagespace.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/pdfedit.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/propedit_all.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/propedit_catalog.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/propedit_edit.png"
rm -f -r "/usr/local/share/doc/pdfedit/gui/images/propedit_edit_add.png"
...
"
```

## 5. Las bases del Makefile

## a. Bases

El programa **make** utiliza un fichero Makefile para ejecutar un conjunto de acciones como la compilación de un proyecto, pero este fichero no se limita a eso: se trata de una especie de script por niveles.

Supongamos el proyecto siguiente, encargado de mostrar «Hola». Muchas cosas para un ejemplo tan limitado, pero se trata de un ejemplo.

```
$ cat hola.h
#ifndef H_HOLA
#define H_HOLA
void Hola(void);
#endif

$ cat hola.c
#include <stdio.h>
#include <stdlib.h>

void Hola(void)
{
    printf("Hola\n");
}

$ cat main.c
#include <stdio.h>
#include <stdlib.h>
#include "hola.h"

int main(void)
{
    Hola();
    return 0;
}
```

Para compilar este proyecto a mano, debe ejecutar las etapas siguientes:

```
$ gcc -o hola.o -c hola.c
$ gcc -o hola hola.o main.o
$ gcc -o main.o -c main.c
$ ./hola
Hola
```

Makefile se compone de reglas con la estructura siguiente:

```
meta: dependencia
comandos
```

Por lo tanto, un primer Makefile podría ser:

```
$ cat Makefile
hola: hola.o main.o
    gcc -o hola hola.o main.o

hola.o: hola.c
    gcc -o hola.o -c hola.c

main.o: main.c hola.h
    gcc -o main.o -c main.c
```

Primera regla: para ejecutar la regla **hola**, hace falta disponer de los ficheros **hola.o** y **main.o**. Si se dispone

de ellos, hay que ejecutar el comando **gcc -o hola hola.o main.o**.

Segunda regla: para ejecutar la regla `hola.o` hace falta disponer del fichero `hola.c`. Si está presente, entonces se ejecuta el comando **gcc -o hola.o -c hola.c**.

Tercera regla: para ejecutar la regla `main.o` hace falta disponer de los ficheros `main.c` y `hola.h`. Si están presentes, entonces se ejecuta el comando **gcc -o main.o -c main.c**.

Las dos últimas reglas permiten resolver la primera. Si usted ejecuta el comando **make**, éste va a determinar cuáles son las reglas aplicables, en qué orden y luego va a aplicarlas en el orden de las dependencias. Si los ficheros están actualizados, **make** no vuelve a construirlos a no ser que hayan sido modificados.

```
$ rm -f *.o
$ make
gcc -o hola.o -c hola.c
gcc -o main.o -c main.c
gcc -o hola hola.o main.o
$ ./hola
Hola
```

## b. Makefile intermedio

El anterior Makefile funciona, pero no es óptimo:

- No permite compilar varios binarios.
- No permite limpiar los ficheros temporales (.o) después de la compilación.
- No permite forzar la compilación del proyecto.

El añadido de nuevas reglas permite paliar estos problemas:

- all: genera n reglas;
- clean: limpia los .o;
- mrproper: llama a clean y suprime los binarios.

```
$ cat Makefile
all: hola

hola: hola.o main.o
      gcc -o hola hola.o main.o

hola.o: hola.c
      gcc -o hola.o -c hola.c

main.o: main.c hola.h
      gcc -o main.o -c main.c

clean:
      rm -rf *.o

mrproper: clean
      rm -rf hola

$ make clean
rm -rf *.o
$ make mrproper
rm -rf *.o
rm -rf hola
```

```
$ make all
gcc -o hola.o -c hola.c
gcc -o main.o -c main.c
gcc -o hola hola.o main.o
```

## c. Un poco más complejo

### Variables de usuario

Para terminar esta pequeña presentación, puede definir variables en su fichero y utilizar variables internas predefinidas.

El Makefile se convierte en:

```
$ cat Makefile
CC=gcc
CFLAGS=-W -Wall -ansi -pedantic
LDFLAGS=
EXEC=hola

all: $(EXEC)

hola: hola.o main.o
    gcc -o hola hola.o main.o $(LDFLAGS)

hola.o: hola.c
    gcc -o hola.o -c hola.c $(CFLAGS)

main.o: main.c hola.h
    gcc -o main.o -c main.c $(CFLAGS)

clean:
    rm -rf *.o

mrproper: clean
    rm -rf $(EXEC)
```

### Variables internas

Entre las variables internas:

- \$@: nombre de la meta.
- \$<: nombre de la primera dependencia.
- \$^: lista de las dependencias.
- \$?: dependencias más recientes que la actual.
- \$\*: nombre del fichero sin el sufijo.

El Makefile se convierte en:

```
$ cat Makefile
CC=gcc
CFLAGS=-W -Wall -ansi -pedantic
LDFLAGS=
EXEC=hola

all: $(EXEC)
```

```

hola: hola.o main.o
      gcc -o $@ $^ $(LDFLAGS)

hola.o: hola.c
      gcc -o $@ -c $< $(CFLAGS)

main.o: main.c hola.h
      gcc -o $@ -c $< $(CFLAGS)

clean:
      rm -rf *.o

mrproper: clean
      rm -rf $(EXEC)

```

### **Reglas de inferencia**

Existen reglas predefinidas, basadas en accesos directos, que permiten generar metas en función del nombre del fichero C y objeto: **%.o: %.c**. La tentación de crear una regla única para **main.o** y **hola.o** es grande:

```

%.o: %.c
      gcc -o $@ -c $< $(CFLAGS)

```

Esta regla es correcta, pero falta la dependencia del header **hola.h**: si se modifica éste, ya no se compila más el proyecto. Hay que añadir una regla específica:

```
El
main.o: hola.h
```

Makefile se convierte en:

```

$ cat Makefile
CC=gcc
CFLAGS=-W -Wall -ansi -pedantic
LDFLAGS=
EXEC=hola

all: $(EXEC)

hola: hola.o main.o
      gcc -o $@ $^ $(LDFLAGS)

%.o: %.c
      gcc -o $@ -c $< $(CFLAGS)

main.o: hola.h

clean:
      rm -rf *.o

mrproper: clean
      rm -rf $(EXEC)

```

# Gestionar las librerías compartidas

## 1. Fundamentos

Una librería compartida es un fichero particular que contiene una lista de funciones, o API, accesible a cualquier programa que lo necesite sin tener que volver a escribirlas. Al contrario de lo que sucede con las librerías estáticas, el programa accede de manera dinámica a las funciones situadas en un fichero aparte. N programas diferentes pueden acceder a las funciones propuestas por la librería. Las librerías agrupan funciones propias de un dominio o un conjunto de dominios coherentes: tratamiento de imágenes, sonido, acceso a una base de datos, etc.

Un conjunto de funciones propuestas por una o varias librerías compartidas forma una **API**, *Application Programming Interface*, y a veces se agrupan en un framework que ofrece una solución completa para un dominio dado.

Se establece un vínculo entre el programa y una librería compartida durante la etapa de la edición de los vínculos por el editor de vínculos **ld**, al cual puede llamar el compilador **gcc** con la opción **-I<lib>**.

Otra posibilidad para un programa consiste en utilizar la función C **dlopen**, que abre una librería dinámica como un fichero y que accede a las funciones contenidas mediante punteros de funciones.

Si un programa depende de una librería compartida y ésta está ausente, el programa no podrá funcionar nunca más.

Bajo Linux (y Unix en general), las librerías compartidas se llaman **Shared Objects** (so) en el sentido en el que se trata de ficheros de objetos sin bloque de instrucción **main**. Llevan el sufijo **.so**.

Una librería puede disponer de varias versiones, que pueden ser o no compatibles, y se puede precisar la versión durante la edición de los vínculos. También es posible determinar una versión por defecto.

## 2. Lugar de almacenamiento

Por convención se colocan las librerías compartidas en directorios llamados lib:

- **/lib**: librerías de sistema básicas, vitales;
- **/usr/lib**: librerías de usuario básicas, no necesarias al boot;
- **/usr/local/lib**: librerías locales para los programas para la máquina;
- **/usr/X11R6/lib**: librerías del entorno X Window;
- **/opt/kde4/lib**: librerías de KDE...

```
$ ls -l /lib
total 6024
...
-rwxr-xr-x 1 root root 114636 oct 23 2007 ld-2.6.1.so
lrwxrwxrwx 1 root root      11 oct  5 2007 ld-linux.so.2 -> ld-2.6.1.so
lrwxrwxrwx 1 root root      13 oct  5 2007 ld-lsb.so.2 -> ld-linux.so.2
lrwxrwxrwx 1 root root      13 oct  5 2007 ld-lsb.so.3 -> ld-linux.so.2
lrwxrwxrwx 1 root root      15 oct  5 2007 libacl.so.1 -> libacl.so.1.1.0
-rw xr-xr-x 1 root root  27864 sep 22 2007 libacl.so.1.1.0
lrwxrwxrwx 1 root root      15 oct  5 2007 libaio.so.1 -> libaio.so.1.0.1
-rw xr-xr-x 1 root root   5248 sep 21 2007 libaio.so.1.0.1
-rw xr-xr-x 1 root root  10256 oct 23 2007 libanl-2.6.1.so
lrwxrwxrwx 1 root root      15 oct  5 2007 libanl.so.1 -> libanl-2.6.1.so
```

```
lrwxrwxrwx 1 root root      20 oct  5 2007 libapparmor.so.1 ->
libapparmor.so.1.0.2
-rw-r-xr-x 1 root root 30404 sep 22 2007 libapparmor.so.1.0.2
lrwxrwxrwx 1 root root      16 oct  5 2007 libattr.so.1 ->
libattr.so.1.1.0
-rw-r--r-- 1 root root 18272 sep 21 2007 libattr.so.1.1.0
...
```

La librería más importante del sistema es la C. Todos los programas compilados son vinculados a libc. Basta suprimir este fichero (un error de principiante) para hacer caer todo el sistema.

```
$ ls -l libc.so.6
lrwxrwxrwx 1 root root 13 oct  5 2007 libc.so.6 -> libc-2.6.1.so
```

Los directorios de las librerías contienen muchos vínculos simbólicos. Estos vínculos están aquí, entre otras cosas, para gestionar las versiones y la compatibilidad entre las versiones (como sucede, por ejemplo, cuando dos versiones coexisten).

```
$ cd /usr/lib
$ ls -l libXm.*
lrwxrwxrwx 1 root root      14 oct 17 2007 libXm.so.3 -> libXm.so.3.0.3
-rw-r-xr-x 1 root root 2371164 oct 12 2007 libXm.so.3.0.3
lrwxrwxrwx 1 root root      14 oct  5 2007 libXm.so.4 -> libXm.so.4.0.0
-rw-r-xr-x 1 root root 2496528 sep 22 2007 libXm.so.4.0.0
```

### 3. ¿Qué librerías vinculadas hay?

El comando **ldd** permite determinar cuáles son las librerías vinculadas a un programa, y también si están presentes o no.

```
$ ldd pdfedit
    linux-gate.so.1 => (0xfffffe000)
    libfreetype.so.6 => /usr/lib/libfreetype.so.6 (0xb7ea5000)
    libz.so.1 => /lib/libz.so.1 (0xb7e92000)
    libt1.so.5 => /usr/lib/libt1.so.5 (0xb7e3d000)
    libqt-mt.so.3 => /usr/lib/libqt-mt.so.3 (0xb7739000)
    libXext.so.6 => /usr/lib/libXext.so.6 (0xb772a000)
    libX11.so.6 => /usr/lib/libX11.so.6 (0xb760f000)
    libstdc++.so.6 => /usr/lib/libstdc++.so.6 (0xb7520000)
    libm.so.6 => /lib/libm.so.6 (0xb74fb000)
    libgcc_s.so.1 => /lib/libgcc_s.so.1 (0xb74ef000)
    libc.so.6 => /lib/libc.so.6 (0xb73bc000)
    libpthread.so.0 => /lib/libpthread.so.0 (0xb73a5000)
    libpng12.so.0 => /usr/lib/libpng12.so.0 (0xb7380000)
    libXi.so.6 => /usr/lib/libXi.so.6 (0xb7376000)
    libXrender.so.1 => /usr/lib/libXrender.so.1 (0xb736d000)
    libXrandr.so.2 => /usr/lib/libXrandr.so.2 (0xb7366000)
    libXcursor.so.1 => /usr/lib/libXcursor.so.1 (0xb735c000)
    libXinerama.so.1 => /usr/lib/libXinerama.so.1 (0xb7358000)
    libXft.so.2 => /usr/lib/libXft.so.2 (0xb7345000)
    libfontconfig.so.1 => /usr/lib/libfontconfig.so.1 (0xb7318000)
    libSM.so.6 => /usr/lib/libSM.so.6 (0xb730f000)
    libICE.so.6 => /usr/lib/libICE.so.6 (0xb72f6000)
    libdl.so.2 => /lib/libdl.so.2 (0xb72f2000)
    libXau.so.6 => /usr/lib/libXau.so.6 (0xb72ee000)
    libxcb-xlib.so.0 => /usr/lib/libxcb-xlib.so.0 (0xb72ea000)
    libxcb.so.1 => /usr/lib/libxcb.so.1 (0xb72d1000)
    /lib/ld-linux.so.2 (0xb7f3c000)
    libXfixes.so.3 => /usr/lib/libXfixes.so.3 (0xb72cb000)
    libexpat.so.1 => /lib/libexpat.so.1 (0xb72aa000)
```

Pongamos ahora por caso que falte una librería (ha sido movido voluntariamente para poder hacer esta demostración):

```
$ ldd /usr/bin/esd
    linux-gate.so.1 => (0xfffffe000)
    libwrap.so.0 => /lib/libwrap.so.0 (0xb7f6d000)
libesd.so.0 => not found
    libasound.so.2 => /usr/lib/libasound.so.2 (0xb7eb1000)
    libaudiofile.so.0 => /usr/lib/libaudiofile.so.0 (0xb7e8e000)
    libm.so.6 => /lib/libm.so.6 (0xb7e69000)
    libc.so.6 => /lib/libc.so.6 (0xb7d36000)
    libdl.so.2 => /lib/libdl.so.2 (0xb7d31000)
    libpthread.so.0 => /lib/libpthread.so.0 (0xb7d1a000)
    /lib/ld-linux.so.2 (0xb7f9f000)
```

Falta la librería `libesd.so.0`. Es imposible ejecutar el programa:

```
$ ./esd
./esd: error while loading shared libraries: libesd.so.0: cannot
open shared object file: No such file or directory
```

## 4. Configurar la caché del editor de vínculos

La edición de los vínculos con una librería compartida es dinámica y se hace en el momento de la ejecución del programa por el sistema con la ayuda de la librería **ld.so**. El binario proporciona el nombre de las librerías a vincular con la ejecución, pero no la ruta. Las funciones de **ld.so** determinan la librería a utilizar en función de su nombre, entre las rutas que conocen.

Cualquier programa se vincula a la librería **ld.so** o más bien **ld-linux.so** (`ld-linux.so.2`).

El cargador de vínculos **ld.so** busca las librerías en varios lugares, entre los cuales, se encuentran, por este orden:

- las rutas precisadas en la variable de entorno **LD\_LIBRARY\_PATH**. Se separan las rutas, como para PATH, por ":" ;
- el contenido del fichero `/etc/ld.so.cache`, que contiene una lista compilada (formato binario) de las librerías encontradas en las rutas predefinidas;
- los directorios `/lib` y `/usr/lib`.

La búsqueda en `/lib` y `/usr/lib` está implícita. Por otro lado, el hecho de rellenar la variable **LD\_LIBRARY\_PATH** no impide la búsqueda de las librerías en otros sitios, si no está en una de las rutas de la lista.

Para evitar la instalación de una variable cuyo contenido se puede manejar con dificultad, **ld.so** propone una caché que usted puede modificar por sí mismo. Se construye la caché desde el contenido del fichero `/etc/ld.so.conf` y del comando **ldconfig**.

Este fichero contiene la lista de los directorios que contienen las librerías compartidas:

```
# cat /etc/ld.so.conf
/usr/X11R6/lib/Xaw3d
/usr/X11R6/lib
/usr/lib/Xaw3d
/usr/i386-suse-linux/lib
/usr/local/lib
/opt/kde3/lib
include /etc/ld.so.conf.d/*.conf
```

En vez de modificar este fichero, un paquete o usted mismo puede decidir añadir un fichero en `/etc/ld.so.conf.d` que contenga la ruta o las rutas de sus nuevas librerías.

No basta con añadir la ruta: debe regenerar la caché con el comando **ldconfig**.

```
# ldconfig
```

comando **ldconfig**:

- actualiza la caché para las rutas definidas en `/etc/ld.so.conf` y asociadas, así como para `/usr/lib` y `/lib`;
- actualiza los vínculos simbólicos en las librerías;
- permite también listar las librerías conocidas en la caché.

Se aceptan las siguientes opciones:

Opción	Papel
-v	Modo verbose: indica lo que <b>ldconfig</b> efectúa
-N	No vuelve a construir la caché
-X	No actualiza los vínculos
-p	Lista el contenido de la caché

Para listar las librerías conocidas del editor de vínculos utilice:

```
# ldconfig -p
1940 libs encontradas en la caché «/etc/ld.so.cache»
    libzyppp.so.324 (libc6) => /usr/lib/libzyppp.so.324
    libzvbi.so.0 (libc6) => /usr/lib/libzvbi.so.0
    libzvbi-chains.so.0 (libc6) => /usr/lib/libzvbi-chains.so.0
    libzip.so.1 (libc6) => /usr/lib/libzip.so.1
    libzio.so.0 (libc6) => /usr/lib/libzio.so.0
    libz.so.1 (libc6) => /lib/libz.so.1
    libz.so (libc6) => /usr/lib/libz.so
    liby2util.so.3 (libc6) => /usr/lib/liby2util.so.3
    liby2storage.so.2 (libc6) => /usr/lib/liby2storage.so.2
    liby2.so.2 (libc6) => /usr/lib/liby2.so.2
    libycpvalues.so.3 (libc6) => /usr/lib/libycpvalues.so.3
    libycp.so.3 (libc6) => /usr/lib/libycp.so.3
    libx264gtk.so.54 (libc6) => /usr/lib/libx264gtk.so.54
    libx264gtk.so (libc6) => /usr/lib/libx264gtk.so
...
...
```

Para saber lo que haría **ldconfig** pero sin actualizar nada, use:

```
# ldconfig -N -X -v
/usr/X11R6/lib:
    libfglrx_pp.so.1.0 -> libfglrx_pp.so.1.0
    libfglrx_gamma.so.1 -> libfglrx_gamma.so.1.0
    libfglrx_tvout.so.1 -> libfglrx_tvout.so.1.0
    libGL.so.1 -> libGL.so.1.2
/usr/local/lib:
/opt/kde3/lib:
```

```
libkdeinit_ksmserver.so -> libkdeinit_ksmserver.so  
libkdeinit_klipper.so -> libkdeinit_klipper.so  
libkdeinit_kcminit.so -> libkdeinit_kcminit.so  
libkdetvvideo.so.0 -> libkdetvvideo.so.0.0.0  
libkmailprivate.so -> libkmailprivate.so  
...
```

Finalmente, para actualizar y ver el resultado use:

```
# ldconfig -v
```

(la salida es la misma que la del comando anterior).

# Requisitos y objetivos

## 1. Requisitos

- Tener una versión cualquiera de Linux.
- Disponer de un terminal (consola o gráfico).

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Trabajar con la línea de comandos.
- Manejar el historial.
- Efectuar operaciones básicas en el sistema de ficheros.
- Buscar ficheros en el árbol.
- Trabajar en ficheros de texto con vi.
- Instalar redirecciones y tuberías.
- Efectuar búsquedas en ficheros texto.
- Utilizar los filtros y utilidades.
- Gestionar los procesos.
- Modificar la ruta de búsqueda, las variables y el entorno.
- Modificar la configuración del bash.
- Aprender a crear scripts.
- Efectuar pruebas y evaluar expresiones.
- Utilizar estructuras de control.
- Crear funciones.

# Las variables

Se distinguen tres tipos: usuario, sistema y especiales. El principio consiste en poder asignar un contenido a un nombre de variable, en general una cadena de caracteres o valores numéricos.

## 1. Nomenclatura

Un nombre de variables obedece a ciertas reglas:

- Se puede componer de letras minúsculas, mayúsculas, cifras, caracteres de subrayado.
- El primer carácter no puede ser una cifra.
- El tamaño de un nombre suele ser ilimitado (pero no hay que pasarse tampoco).
- Las convenciones quieren que las variables de usuario estén en minúsculas para diferenciarlas de las variables de sistema. A elección del usuario.

## 2. Declaración y asignación

Se declara una variable en cuanto se le asigna un valor. Se efectúa la asignación con el signo `=`, sin espacio antes ni después del signo.

```
var=Hola
```

## 3. Acceso y visualización

Puede acceder al contenido de una variable colocando el signo `$` delante del nombre de la variable. Cuando el shell encuentra el `$`, intenta interpretar la palabra siguiente como si fuera una variable. Si existe, entonces se sustituye el `$nombre_variable` por su contenido, o por un texto vacío en el caso contrario. Se habla también de referenciado de variable.

```
$ ruta=/tmp/seb
$ ls $ruta
...
$ cd $ruta
$ pwd
/tmp/seb
$ cd $ruta/dir1
$ pwd
/tmp/seb/dir1
```

Para visualizar la lista de las variables, se utiliza el comando `env`. Muestra las variables de usuario y de sistema, nombre y contenido.

```
$ env
LESSKEY=/etc/lesskey.bin
NNTPSERVER=news
INFODIR=/usr/local/info:/usr/share/info:/usr/info
MANPATH=/usr/local/man:/usr/share/man
KDE_MULTIHEAD=false
SSH_AGENT_PID=26377
HOSTNAME=p64p17bicb3
DM_CONTROL=/var/run/xdmctl
```

```
XKEYSYMDB=/usr/share/X11/XKeysymDB
HOST=p64p17bicb3
SHELL=/bin/bash
TERM=xterm
PROFILEREAD=true
HISTSIZE=1000
...
```

Una variable puede contener caracteres especiales, particularmente el espacio. El ejemplo siguiente no funciona:

```
$ c=Hola amigos
: not found
$ echo $c
```

Para hacer que funcione, hay que cerrar los caracteres especiales uno por uno, o colocarlos entre comillas o apóstrofes.

```
c=Hola\ Amigos # Solución pesada
c="Hola amigos" # Solución correcta
c='Hola amigos' # Solución correcta
```

La principal diferencia entre las comillas y los apóstrofes es la interpretación de las variables y de las sustituciones. " y ' se cierran mutuamente.

```
$ a=Julio
$ b=César
$ c="$a $b conquistó la Galia"
$ d='$a $b conquistó la Galia'
$ echo $c
Julio César conquistó la Galia
$ echo $d
$a $b conquistó la Galia
$ echo "Linux es genial"
Linux es genial
$ echo 'Linux "demasiado bien"'
Linux "demasiado bien"
```

## 4. Supresión y protección

Se suprime una variable con el comando **unset**. Puede proteger una variable en modo escritura y contra su supresión con el comando **readonly**. Una variable en modo de sólo lectura, incluso vacía, es exclusiva. No existe ningún medio de sustituirla en escritura y de suprimirla, salvo saliendo del shell.

```
$ a=Julio
$ b=César
$ echo $a $b
Julio César
$ unset b
$ echo $a $b
Julio
$ readonly a
$ a=Nerón
a: is read only
$ unset a
a: is read only
```

## 5. Export

Por defecto una variable es accesible únicamente desde el shell donde ha sido definida. El siguiente ejemplo declara la variable **a** en el entorno del shell actual y luego trata de visualizar su valor a través de un script lanzado desde este mismo shell. El script no reconoce la variable **a**: no se visualiza nada.

```
$ a=Julio  
$ echo 'echo "a=$a"' > ver_a.sh  
$ chmod u+x ver_a.sh  
$ ./ver_a.sh  
a=
```

comando **export** permite exportar una variable de manera que su contenido sea visible por los scripts y otros subshells. Se pueden modificar las variables exportadas en el script, pero estas modificaciones sólo se aplican al script o al subshell. Esta vez, el primer script puede acceder a la variable **a** exportada. Pero las modificaciones siguen locales en el script. Una vez éste ha terminado, la modificación desaparece.

```
$ export a  
$ ./ver_a.sh  
a=Julio  
$ echo 'a=Nerón ; echo "a=$a"' >> ver_a.sh  
$ ./ver_a.sh  
a=Julio  
a=Nerón  
$ echo $a  
Julio
```

## 6. Llaves

Las llaves básicas {} permiten identificar el nombre de una variable. Imaginemos la variable fichero que contiene el nombre de fichero «lista». Usted quiere copiar lista1 en lista2.

```
$ fichero=list  
$ cp $ficherol $fichero2  
cp: operando fichero que falta  
Para saber más, haga: «cp --help».
```

No funciona, ya que no se interpreta \$fichero sino \$ficherol y \$fichero2, que no existen.

```
$ cp ${fichero}2 ${fichero}1
```

En este caso, esta línea equivale a:

```
$ cp lista2 listal
```

## 7. Llaves y sustitución condicional

Las llaves disponen de una sintaxis particular.

```
{variable:Sustitución}
```

Según el valor o la presencia de la variable, es posible sustituir su valor por otro.

Sustitución	Significado
{x:-texto}	Si la variable x está vacía o es inexistente, el texto cogerá su sitio. En caso contrario, es el contenido de la variable el que prevaldrá.

{x:=texto}	Si la variable x está vacía o es inexistente, el texto cogerá su sitio y se convertirá en el valor de la variable.
{x:+texto}	Si la variable x está definida y no vacía, el texto cogerá su sitio. En el caso contrario, una cadena vacía coge su lugar.
{x:?texto}	Si la variable x está vacía o es inexistente, se interrumpe el script y se visualiza el mensaje de texto. Si el texto está ausente, se visualiza un mensaje de error estándar.

```
$ echo $nombre
$ echo ${nombre:-Juan}
Juan
$ echo $nombre
$ echo ${nombre:=Juan}
Juan
$ echo $nombre
Juan
$ echo ${nombre:+Valor definido}
Valor definido
$ unset nombre
$ echo ${nombre:?Variable ausente o no definida}
nombre: Variable ausente o no definida
$ nombre=Juan
$ echo ${nombre:?Variable ausente o no definida}
Juan
```

## 8. Variables de sistema

Además de las variables que el usuario puede definir por sí mismo, se inicia el shell con un cierto número de variables predefinidas útiles para ciertos comandos y accesibles por el usuario. Se puede modificar el contenido de estas variables de sistema, pero hay que ser cuidadoso, ya que algunas tienen una incidencia directa sobre el comportamiento del sistema.

Variable	Contenido
HOME	Ruta de acceso del directorio de usuario. Directorio por defecto en caso de uso de CD.
PATH	Lista de directorios, separados por :, donde el shell va a buscar los comandos externos y otros scripts y binarios. La búsqueda se hace en el orden de los directorios insertados.
PS1	Prompt String 1, cadena que representa el prompt estándar visualizado en la pantalla por el shell en espera de inserción de comando.
PS2	Prompt String 2, cadena que representa un prompt secundario en caso de que se deba completar la inserción.
IFS	Internal Field Separator, lista de los caracteres que separan las palabras en una línea de comandos. Por defecto se trata del espacio, de la tabulación y del salto de línea.

MAIL	Ruta y fichero que contiene los mensajes del usuario.
SHELL	Ruta completa del shell en curso de ejecución.
LANG	Definición del idioma a utilizar, así como del juego de caracteres.
USER	Nombre del usuario en curso.
LOGNAME	Nombre del login utilizado durante la conexión.
HISTFILE	Nombre del fichero del historial, en general \$HOME/.sh_history.
HISTSIZE	Tamaño en número de líneas del historial.
OLDPWD	Ruta de acceso del directorio al que se ha accedido anteriormente.
PS3	Define la línea de comandos para escribir un select.
PWD	Ruta actual.
RANDOM	Genera y contiene un número aleatorio entre 0 y 32767.

## 9. Variables especiales

Se trata de variables accesibles únicamente para lectura y cuyo contenido suele ser contextual.

Variable	Contenido
\$?	Código de retorno del último comando ejecutado.
\$\$	PID del shell activo.
\$!	PID del último proceso iniciado en segundo plano.
\$-	Las opciones del shell.

```
$ echo $$  
23496  
$ grep memoria lista  
$ echo $?  
1  
$ grep ratón lista  
ratón óptico 30      15  
$ echo $?  
0  
$ ls -lR >pepito.txt 2>&1 &  
26675  
$ echo $!  
26675
```

## 10. Longitud de una cadena

Es posible obtener la longitud de una cadena con el carácter #.

```
$ a=Julio  
$ echo "Longitud de $a: ${#a}"  
Longitud de Julio: 5
```

## 11. Tablas y campos

Linux cuenta con dos medios para declarar una tabla: la utilización de los corchetes [ ] y la creación global. El primer elemento es 0; el último, 1023. Para acceder al contenido de la tabla, hay que poner la variable Y el elemento entre llaves {}.

```
$ Nombre[0]="Julio"  
$ Nombre[1]="Román"  
$ Nombre[2]="Francisco"  
$ echo ${Nombre[1]}  
Román
```

```
$ Nombre=(Julio Román Francisco)  
$ echo ${nombre[2]}  
Francisco
```

Para listar todos los elementos:

```
$ echo ${Nombre[*]}  
Julio Román Francisco
```

Para conocer el número de elementos:

```
$ echo ${#Nombre[*]}  
3
```

índice es una variable, no se pone \$ delante:

```
$ idx=0  
$ echo ${Nombre[idx]}  
Julio
```

## 12. Variables tipadas

Las variables pueden ser de tipo numérico entero (integer) con el comando **typeset -i**. La ventaja es que permite efectuar cálculos y comparaciones sin pasar por expr. El comando **let** o **((...))** permite cálculos sobre variables.

Operador	Papel
+ - * /	Operaciones sencillas
%	Módulo
< > <= >=	Comparaciones, 1 si verdadero, 0 si falso

<code>== !=</code>	Igual o diferente
<code>&amp; &amp;   </code>	Comparaciones relacionadas con un operador lógico
<code>&amp;   ^</code>	Lógico binario AND OR XOR

```
$ typeset -i resultado
$ resultado=6*7
$ echo $resultado
42
$ resultado=Error
ksh: Error: bad number
$ resultado=resultado*3
126
$ typeset -i add
$ add=5
$ let resultado=add+5 resultado=resultado*add
$ echo $resultado
50
```

# Configuración del bash

## 1. Ficheros de configuración

Se puede lanzar el shell bash en varios modos:

- shell interactiva de conexión (login shell);
- shell interactiva simple;
- shell no interactiva;
- el modo sh;
- etc.

Según su modo de lanzamiento, el shell va a buscar y ejecutar varios scripts y ficheros de configuración. Un fichero de configuración es un script de shell, una secuencia de comandos individuales que tiene como meta configurar el entorno del usuario.

### a. Shell de conexión

Se lanza el shell de conexión después de la inserción del login y de la contraseña en la consola. Corresponde al que se ha precisado al final de cada línea de /etc/passwd. En este modo, el shell busca ejecutar, en este orden y si están presentes:

- /etc/profile
- ~/.bash\_profile
- ~/.bash\_login
- ~/.profile

En el momento de desconectarse, intenta ejecutar:

- - ~/.bash\_logout

### b. Shell simple

El shell interactivo simple corresponde a la ejecución del bash en una ventana (xterm, konsole), una consola o manualmente (teclar bash en una consola). En este caso, sólo se ejecutará el fichero siguiente si existe:

- ~/.bashrc

---

 Observe que en muchas distribuciones Linux, .bashrc es llamado o bien por .bash\_profile, o bien por etc/profile y que por lo tanto es la configuración ubicada en .bash\_profile la que siempre será invocada.

---

### c. Modo Bourne shell

Cuando se inicia el bash en modo Bourne Shell mediante el comando **sh**, en shell de conexión o no, intenta ejecutar los ficheros en este orden:

- /etc/profile

- `~/.profile`

#### d. Modo no interactivo

Se puede iniciar el shell en modo interactivo. Suele ser el caso cuando usted ejecuta un script. En este caso, no hay ningún script ejecutado por defecto en el momento del inicio salvo si usted especifica una variable llamada `BASH_ENV` que contiene la ruta de un script. En este caso bash carga y ejecuta este fichero antes de comenzar con la ejecución del script o del comando.

## 2. Comandos set

El shell dispone de una decenas de opciones; la mayoría de ellas se pueden configurar con la ayuda del comando **set**. Las que siguen representan un mero resumen. El `-` antes de una opción la activa. Un `+` la desactiva.

Opción	Resultado
<code>-a / -o allexport</code>	Se exportarán todas las variables de manera automática.
<code>-u / -o nounset</code>	Por defecto, el shell trata las variables inexistentes como cadenas vacías. Esta opción produce un error si la variable no existe.
<code>-x / -o xtrace</code>	Muestra todos los comandos a medida que se ejecutan: ideal al principio del script para depuración.
<code>-o vi</code>	Uso de la línea de comandos con la sintaxis de vi.
<code>-o emacs</code>	Uso de la línea de comandos con la sintaxis de emacs.
<code>-C / -o noclobber</code>	Prohibe las redirecciones en salida si el fichero ya existe.
<code>history</code>	Autoriza la gestión del historial.

El manual del shell le proporcionará todas las opciones posibles.

# Programación shell

## 1. Estructura y ejecución de un script

El shell no es solamente un simple intérprete de comandos, sino que dispone de un verdadero lenguaje de programación; en particular, con gestión de las variables, control del flujo y bucles, operaciones sobre las variables, funciones...

Se agrupan todas las instrucciones y los comandos dentro de un script. Durante su ejecución, cada línea se leerá una por una y se ejecutará. Una línea puede componerse de comandos internos o externos, de comentarios o estar vacía. Es posible diseñar varias instrucciones consecutivas siempre separadas por ; o relacionadas de manera condicional por && o ||. El ; es el equivalente de un **salto de línea**.

Por convención, los nombres de los scripts del shell terminan en general (no es obligatorio) por «sh» para el Bourne Shell y el Bourne Again Shell; «.ksh» para el Korn Shell y «csh» para el C Shell.

Para que un script sea ejecutable directamente: \$ chmod u+x miscript

Para ejecutarlo: \$ ./miscript

Para evitar el .:/

```
$ PATH=$PATH:  
$ miscript
```

Observe que se coloca el punto en última posición en el PATH. Ponerlo en primera posición puede presentar un riesgo para la seguridad: se ha colocado un nuevo comando ls modificado en su directorio.

Cuando se inicia un script, se crea un nuevo shell hijo que va a ejecutar cada uno de los comandos. Si es un comando interno, el nuevo shell lo ejecuta directamente. Si es un comando externo binario, se creará un nuevo hijo para ejecutarlo. En el caso de un script de shell, se inicia un nuevo shell hijo para leer este nuevo shell línea por línea.

Una línea de comentario siempre empieza con el carácter #. Se puede colocar un comentario al final de una línea que ya comporta comandos.

```
# La línea siguiente efectúa un ls  
ls # La línea en cuestión
```

La primera línea reviste una importancia particular, ya que permite especificar qué shell va a ejecutar el script:

```
#!/bin/bash  
#!/bin/ksh
```

En el primer caso, es un script Bourne Again; en el otro, un script Korn.

## 2. Argumentos de un script

### a. Parámetros de posición

Los parámetros de posición son también variables especiales utilizadas cuando se pasan parámetros a un script.

Variable	Contenido
\$0	Nombre del comando (del script).
\$1-9	\$1,\$2,\$3... Los nueve primeros parámetros pasados al script.
\$#	Número total de parámetros pasados al script.
\$*	Lista de todos los parámetros en formato "\$1 \$2 \$3 ...".
\$@	Lista de los parámetros en forma de elementos distintos "\$1" "\$2" "\$3" ...

```
$ cat param.sh
#!/bin/bash

echo "Nombre: $0"
echo "Número de parámetros: $#"
echo "Parámetros: 1=$1 2=$2 3=$3"
echo "Lista: $*"
echo "Elementos: $@"

$ param.sh juanito jorgito jaimito
Nombre: ./param.sh
Número de parámetros: 3
Parámetros: 1=juanito 2=jorgito 3=jaimito
Lista: juanito jorgito jaimito
Elementos: juanito jorgito jaimito
```

La diferencia entre \$@ y \$\* no salta a la vista. Retome el ejemplo anterior con una pequeña modificación:

```
$ param.sh juanito "jorgito jaimito"
Nombre: ./param.sh
Número de parámetros: 2
Parámetros: 1=juanito 2=jorgito jaimito 3=
Lista: juanito jorgito jaimito
Elementos: juanito jorgito jaimito
```

Esta vez, sólo se pasan dos parámetros. Sin embargo, las listas parecen visualmente idénticas. En realidad, si la primera contiene, en efecto:

"juanito jorgito jaimito"

La segunda contiene:

"juanito" "jorgito jaimito"

O sea, dos elementos. En el primer ejemplo, tenía:

"juanito" "jorgito" "jaimito"

## b. Redefinición de los parámetros

Además de listar las variables, la instrucción **set** permite redefinir el contenido de las variables de posición. Con:

set valor1 valor2 valor3 ...

\$1 tomará como contenido valor1; \$2, valor2, y así sucesivamente.

```
$ cat param2.sh
#!/bin/bash
echo "Antes:"
echo "Número de parámetros: $#"
echo "Parámetros: 1=$1 2=$2 3=$3 4=$4"
echo "Lista: $*"
set alfred oscar romeo zulu
echo "después set alfred oscar romeo zulu"
echo "Número de parámetros: $#"
echo "Parámetros: 1=$1 2=$2 3=$3 4=$4"
echo "Lista: $*"

$ ./param2.sh juanito jorgito jaimito donald gilito
Antes:
Número de parámetros: 5
Parámetros: 1=juanito 2=jorgito 3=jaimito 4=donald
Lista: juanito jorgito jaimito donald gilito
después set alfred oscar romeo zulu
Número de parámetros: 4
Parámetros: 1=alfred 2=oscar 3=romeo 4=zulu
Lista: alfred oscar romeo zulu
```

### c. Reorganización de los parámetros

El comando **shift** permite modificar la posición de los parámetros. Una simple llamada desplaza todos los parámetros una posición, suprimiendo el primero: \$2 se convierte en \$1, \$3 se convierte en \$2, y así sucesivamente. El \$1 original desaparece. \$# , \$\* y \$@ se vuelven a definir en consecuencia.

El comando **shift** seguido de un valor n efectúa un desplazamiento de n elementos. Así, con shift 4 \$5 se convierte en \$1, \$6 se convierte en \$2...

```
$ cat param3.sh
#!/bin/bash
set alfred oscar romeo zulu
echo "set alfred oscar romeo zulu"
echo "Número de parámetros: $#"
echo "Parámetros: 1=$1 2=$2 3=$3 4=$4"
echo "Lista: $*"
shift
echo "Después de un shift"
echo "Número de parámetros: $#"
echo "Parámetros: 1=$1 2=$2 3=$3 4=$4"
echo "Lista: $*"

$ ./param3.sh
set alfred oscar romeo zulu
Número de parámetros: 4
Parámetros: 1=alfred 2=oscar 3=romeo 4=zulu
Lista: alfred oscar romeo zulu
Después de un shift
Número de parámetros: 3
Parámetros: 1=oscar 2=romeo 3=zulu 4=
Lista: oscar romeo zulu
```

### d. Salida de script

El comando **exit** permite terminar un script. Por defecto, el valor devuelto es 0 (no error), pero se puede especificar cualquier otro valor de 0 a 255. Puede recuperar el valor de salida mediante la variable \$?.

```
$ exit 1
```

### 3. Entorno del proceso

En principio, solamente las variables exportadas son accesibles por un proceso hijo. Si desea visualizar el contexto relacionado con un hijo (en un script por ejemplo), utilice el comando **env**.

```
$ env
LESSKEY=/etc/lesskey.bin
NNTPSERVER=news
INFODIR=/usr/local/info:/usr/share/info:/usr/info
MANPATH=/usr/local/man:/usr/share/man
KDE_MULTIHEAD=false
SSH_AGENT_PID=28012
HOSTNAME=slyserver
DM_CONTROL=/var/run/xdmctl
XKEYSYMDB=/usr/share/X11/XKeysymDB
HOST=p64p17bicb3
SHELL=/bin/bash
TERM=xterm
PROFILEREAD=true
HISTSIZE=1000
XDM_MANAGED=/var/run/xdmctl/xdmctl-
:0,maysd,mayfn,sched,rsvd,method=classic
XDG_SESSION_COOKIE=16af07a56781b4689718210047060300-
1211264847.394692-546885666
TMPDIR=/tmp
GTK2_RC_FILES=/etc/gtk-2.0/gtkrc:/usr/share/themes//QtCurve/gtk-
2.0/gtkrc:/home/seb/.gtkrc-2.0-qtengine:/home/seb/.gtkrc-
2.0:/home/seb/.kde/share/config/gtkrc-2.0
KDE_NO_IPV6=1
GTK_RC_FILES=/etc/gtk/gtkrc:/home/seb/.gtkrc:/home/seb/.kde/share/co
nfig/gtkrc
GS_LIB=/home/seb/.fonts
WINDOWID=71303176
MORE=-sl
QTDIR=/usr/lib/qt3
XSESSION_IS_UP=yes
KDE_FULL_SESSION=true
GROFF_NO_SGR=yes
JRE_HOME=/usr/lib/jvm/jre
USER=seb
...
```

comando **env** permite volver a definir también el contexto del proceso a ejecutar. Puede ser útil cuando el script debe acceder a una variable que no está presente en el entorno del padre, o que no se desea exportar. La sintaxis es:

```
env var1=valor var2=valor ... comando
```

En el caso de bash, env no es indispensable.

```
var1=valor var2=valor ... comando
```

primera opción es el signo -, entonces se debe suprimir todo el entorno existente para sustituirlo por las nuevas variables y los nuevos valores.

```
$ unset a
$ ./ver_a.sh
```

```
a=
$ env a=jojo ./ver_a.sh
a=jojo
$ echo a=$a
a=
```

## 4. Sustitución de comando

El mecanismo de sustitución permite colocar el resultado de comandos simples o complejos en una variable. Debe situar los comandos a ejecutar entre dos acentos graves «`».

```
$ mi_unix=`uname`
$ echo ${mi_unix}
Linux
$ maquina=`uname -a | cut -d" " -f5`
echo $maquina
SMP
```

Cuidado: sólo se asigna a la variable el canal de salida estándar. El canal de error estándar sale siempre por pantalla en este caso.

Los acentos graves no son siempre ideales para estos tratamientos de comando. En efecto, si trabaja con varios niveles, debe cerrar los que están dentro de los niveles superiores. Asimismo, el bash permite utilizar en su lugar la sintaxis `$(...)`, que no tiene este problema.

```
$ mi_unix=$(uname)
$ echo ${mi_unix}
Linux
$ maquina=$(uname -a | cut -d" " -f5)
echo
$maquina
SMP
```

 No confunda las llaves y los paréntesis. Las primeras aíslan las variables, mientras que los segundos efectúan la sustitución de los comandos.

## 5. El programa test

El programa **test** permite evaluar escenarios de programación. Se puede recuperar el resultado de una condición con la variable `$?` (código retorno). Si el resultado es 0, entonces se ha cumplido la condición.

### a. Pruebas en una cadena

- **test -z "variable"**: cero, retorno OK si la variable está vacía (p. ej.: `test -z "$a"`).
- **test -n "variable"**: no cero, retorno OK si la variable no está vacía (texto cualquiera).
- **test "variable" = cadena**: OK si las dos cadenas son idénticas.
- **test "variable" != cadena**: OK si las dos cadenas son diferentes.

```
$ a=
$ test -z "$a" ; echo $?
0
$ test -n "$a" ; echo $?
1
```

```
$ a=Julio
$ test "$a" = Julio ; echo $?
0
```

Tenga cuidado con colocar correctamente las variables que contienen texto entre comillas, porque si la variable está vacía el sistema producirá un bug:

```
$ a=
$ b=pepito
$ [ $a = $b ] && echo "ok"
bash: [: =: unary operator expected
```

Mientras que

```
[ "$a" = "$b" ] && echo "ok"
```

no produce errores.

## b. Pruebas sobre los valores numéricos

Se deben convertir las cadenas en valores numéricos para poder evaluarlas. Bash sólo gestiona valores enteros. La sintaxis es `test valor1 opción valor2` y las opciones son las siguientes:

Opción	Papel
-eq	Equal: igual
-ne	Not Equal: diferente
-lt	Less than: inferior
-gt	Greater than: superior
-le	Less or equal: inferior o igual
-ge	Greater or equal: superior o igual

```
$ a=10
$ b=20
$ test "$a" -ne "$b" ; echo $?
0
$ test "$a" -ge "$b" ; echo $?
1
$ test "$a" -lt "$b" && echo "$a es inferior a $b"
10 es inferior a 20
```

## c. Pruebas sobre todos los ficheros

La sintaxis es `prueba opción nombre_fichero` y las opciones son las siguientes:

Opción	Papel
-f	Fichero normal.

-d	Un directorio.
-c	Fichero en modo carácter.
-b	Fichero en modo bloque.
-p	Tubería nombrada (named pipe).
-r	Permiso de lectura.
-w	Permiso de escritura.
-x	Permiso de ejecución.
-s	Fichero no vacío (al menos un carácter).
-e	El fichero existe.
-L	El fichero es un vínculo simbólico.
-u	El fichero existe, SUID-Bit activado.
-g	El fichero existe, SGID-Bit activado.

```
$ ls -l
-rw-r--r-- 1 seb  users      1392 Ago 14 15:55 dump.log
lrwxrwxrwx  1 seb  users          4 Ago 14 15:21 vínculo_fic1 -> fic1
lrwxrwxrwx  1 seb  users          4 Ago 14 15:21 vínculo_fic2 -> fic2
-rw-r--r--  1 seb  users        234 Ago 16 12:20 lista1
-rw-r--r--  1 seb  users        234 Ago 13 10:06 lista2
-rwxr--r--  1 seb  users       288 Ago 19 09:05 param.sh
-rwxr--r--  1 seb  users       430 Ago 19 09:09 param2.sh
-rwxr--r--  1 seb  users       292 Ago 19 10:57 param3.sh
drwxr-xr-x  2 seb  users      8192 Ago 19 12:09 dir1
-rw-r--r--  1 seb  users     1496 Ago 14 16:12 resultado.txt
-rw-r--r--  1 seb  users     1715 Ago 16 14:55 pepito.txt
-rwxr--r--  1 seb  users      12 Ago 16 12:07 ver_a.sh
$ test -f vínculo_fic1 ; echo $?
1
$ test -x dump.log ; echo $?
1
$ test -d dir1 ; echo $?
0
```

#### d. Pruebas combinadas por criterios Y, O, NO

Puede efectuar varias pruebas con una sola instrucción. Las opciones de combinación son las mismas que para el comando **find**.

Criterio	Acción
-a	AND, Y lógico
-o	OR, O lógico

!	NOT, NO lógico
(...)	agrupación de las combinaciones. Se deben cerrar los paréntesis \(...\).

```
$ prueba -d "dir1" -a -w "dir1" && echo "dir1: directorio, derecho en escritura"
dir1: directorio, derecho en escritura
```

### e. Sintaxis ligera

Se puede sustituir la palabra test por los corchetes abiertos y cerrados **[...]**. Hay que colocar un espacio antes y otro después de los corchetes.

```
$ [ "$a" -lt "$b" ] && echo "$a es inferior a $b"
10 es inferior a 20
```

El

bash (y el ksh) integra un comando interno que viene a sustituir al programa test. En la práctica, el comando interno es completamente compatible con el comando externo, pero mucho más rápido, ya que no requiere la creación de un nuevo proceso. Para forzar el uso del comando interno, utilice los dobles corchetes **[[...]]**.

```
$ [[ "$a" -lt "$b" ]] && echo "$a es inferior a $b"
10 es inferior a 20
```

## 6. if ... then ... else

La estructura **if then else fi** es una estructura de control condicional.

```
if <comandos_condición>
then
    <comandos ejecutados si condición cumplida>
else
    <comandos ejecutados si última condición no realizada>
fi
```

También puede especificar **elif**, en realidad un else if. Si no se cumple la última condición, se prueba una nueva.

```
$ cat param4.sh
#!/bin/bash
if [ $# -ne 0 ]
then
    echo "$# parametros en linea de comandos"
else
    echo "Ningun parametro; set alfred oscar romeo zulu"
    set alfred oscar romeo zulu
fi

echo "Numero de parametros: $"
echo "Parametros: 1=$1 2=$2 3=$3 4=$4"
echo "Lista: $"

$ ./param4.sh titi pepito
2 parametros en linea de comandos
Numero de parametros: 2
Parametros: 1=pepito 2=titi 3= 4=
Lista: toto titi
```

```
$ ./param4.sh
Ningun parametro; set alfred oscar romeo zulu
Numero de parametros: 4
Parametros: 1=alfred 2=oscar 3=romeo 4=zulu
Lista: alfred oscar romeo zulu
```

## 7. Evaluación múltiple

El comando **case ... esac** permite comprobar el contenido de una variable o de un resultado de manera múltiple.

```
case Valor in
  Modelo1) Comandos ;;
  Modelo2) Comandos ;;
  *) accion_defecto ;;
esac
```

El

modelo es o bien un texto simple o bien uno compuesto de caracteres especiales. Cada bloque de comandos relacionados con el modelo se debe terminar con dos puntos y coma. En cuanto se haya comprobado el modelo, se ejecuta el bloque de comandos correspondiente. El asterisco en la última posición (cadena variable) es la acción por defecto si no se establece ningún criterio; es opcional.

Carácter	Papel
*	Cadena variable (incluso vacía)
?	Un solo carácter
[...]	Un intervalo de caracteres
[!...]	Negación del intervalo de caracteres
	O lógico

```
$ cat case1.sh
#!/bin/bash
if [ $# -ne 0 ]
then
    echo "$# parametros en linea de comandos"
else
    echo "Ningun parametro; set alfred oscar romeo zulu"
    exit 1
fi

case $1 in
  a*)
      echo "Empieza por a"
      ;;
  b*)
      echo "Empieza por b"
      ;;
  fic[123])
      echo "fic1 fic2 o fic3"
      ;;
  *)
      echo "Empieza por cualquier"
      ;;
esac
```

```
exit 0
$ ./case1.sh "adios"
Empieza por a
$ ./case1.sh hola
Empieza por b
$ ./case1.sh fic2
fic1 o fic2 o fic3
$ ./case1.sh error
Enpieza por cualquiera
```

## 8. Inserción del usuario

El comando **read** permite al usuario insertar una cadena y colocarla en una o varias variables. Se valida la inserción con [Intro].

```
read var1 [var2 ...]
```

Si se especifican varias variables, la primera palabra irá en var1; la segunda, en var2, y así sucesivamente. Si hay menos variables que palabras, las últimas palabras van en la última variable.

```
$ cat read.sh
#!/bin/bash
echo "Continuar (S/N)? \c"
read respuesta
echo "repuesta=$respuesta"
case $respuesta in
    S)
        echo "Sí, seguimos"
        ;;
    N)
        echo "No, paramos"
        exit 0
        ;;
    *)
        echo "Error de inserción (S/N)"
        exit 1
        ;;
esac
echo "Usted siguió. Inserte dos palabras o más :\""
read palabra1 palabra2
echo "palabra1=$palabra1\npalabra2=$palabra2"
exit 0
$ ./read.sh
Continuar (S/N)? S
respuesta=S
Sí, seguimos.
Usted siguió. Inserte dos palabras o más: hola amigos
palabra1=hola
palabra2=amigos
```

## 9. Los bucles

Permiten la repetición de un bloque de comandos ya sea un número limitado de veces o bien de manera condicional. Todos los comandos que se deben ejecutar en un bucle se colocan entre los comandos **do** y **done**.

### a. Bucle for

El bucle **for** no se basa en un incremento de valor cualquiera, sino en una lista de valores, ficheros...

```
for var in lista
do
    comandos a ejecutar
done
```

La lista representa un determinado número de elementos que se asignarán a **var** sucesivamente.

### **Con una variable** Lista implícita

```
$ cat for1.sh
#!/bin/bash
for params in $@
do
    echo "$params"
done
$ ./for1.sh test1 test2 test3
test1
test2
test3
```

Si

no especifica ninguna lista a **for**, entonces es la lista de los parámetros la que está implícita. Ningún script anterior hubiera podido parecerse a:

```
for params
do
    echo "$params"
done
```

### **Con una lista de elementos explícita**

Cada elemento colocado después del «in» se utilizará para cada iteración del bucle, uno tras otro.

```
$ cat for2.sh
#!/usr/bin/sh
for params in lista lista2
do
    ls -l $params
done
$ ./for2.sh
-rw-r--r-- 1 oracle system      234 Aug 19 14:09 lista
-rw-r--r-- 1 oracle system      234 Aug 13 10:06 lista2
```

### **Con criterios de búsqueda sobre nombre de ficheros**

Si uno o varios elementos de la lista corresponden a un fichero o a un grupo de ficheros presentes en la posición actual de la estructura, el bucle for considera el elemento como un nombre de fichero.

```
$ cat for3.sh
#!/bin/bash
for params in *
do
    echo "$params \c"
    type_fic=`ls -ld $params | cut -c1`
    case $type_fic in
        -)    echo "Fichero normal" ;;
        d)    echo "Directorio" ;;
        b)    echo "modo bloque" ;;
        l)    echo "vinculo simbolico" ;;
        c)    echo "modo caracter" ;;
```

```

        *)      echo "otro" ;;
esac

done
$ ./for3.sh
casel.sh Fichero normal
dump.log Fichero normal
for1.sh Fichero normal
for2.sh Fichero normal
for3.sh Fichero normal
vinculo_fic1 vinculo simbolico
vinculo_fic2 vinculo simbolico
lista Fichero normal
listal Fichero normal
lista2 Fichero normal
param.sh Fichero normal
param2.sh Fichero normal
param3.sh Fichero normal
param4.sh Fichero normal
read.sh Fichero normal
dir1 Directorio
resultado.txt Fichero normal
pepito.txt Fichero normal
ver_a.sh Fichero normal

```

### **Con un intervalo de valores**

Existen dos métodos para contar de 1 a n con un bucle for. El primero consiste en utilizar una sustitución de comando con el comando **seq**. Su sintaxis básica requiere un parámetro numérico y cuenta de 1 al valor de este parámetro. El manual le mostrará que se puede iniciar en cualquier valor e incrementarse con cualquier valor.

```
$ seq 5
1
2
3
4
5
```

Usándolo con el bucle for, obtenemos este resultado:

```
$ for i in $(seq 5); do echo $i; done
1
2
3
4
5
```

segundo método consiste en utilizar una sintaxis parecida a la del lenguaje C:

```
$ for ((a=1 ; a<=5 ; a++)); do echo $a; done
1
2
3
4
5
```

### **Con una sustitución de comando**

Se puede colocar cualquier comando que produzca una lista de valores a continuación del «in» con la ayuda

de una sustitución de comando. El bucle for tomará el resultado de este comando como lista de elementos con la que iterar.

```
$ cat for4.sh
#!/bin/bash
echo "Lista de los usuarios en /etc/passwd"
for params in `cat /etc/passwd | cut -d: -f1` 
do
    echo "$params "
done
$ ./for4.sh
Lista de los usuarios en /etc/passwd
root
nobody
nobodyV
daemon
bin
uucp
uucpa
auth
cron
lp
tcb
adm
ris
carthic
ftp
stu
...
```

## b. Bucle while

El comando **while** permite un bucle condicional «mientras». Mientras se cumpla la condición, se ejecuta el bloque de comandos. Si la condición ya no es válida, se sale del bucle.

```
while condición
do
    comandos
done
```

```
while
bloque de instrucciones que forman la condición
do
    comandos
done
```

Por ejemplo:

```
$ cat while1.sh
#!/bin/bash
while
    echo "¿Cadena? \c"
    read nombre
    [ -z "$nombre" ]
do
    echo "ERROR: no inserción"
done
echo "Usted insertó: $nombre"
```

## Lectura de un fichero línea por línea

```
#!/bin/bash
cat pepito.txt | while read line
do
    echo "$line"
done
```

```
#!/bin/bash
while read line
do
    echo "$line"
done < pepito.txt
```

Hay una enorme diferencia entre los dos versiones. En la primera, observe la presencia de la tubería (pipe): se ejecuta el bucle en un segundo proceso. ¡Por lo tanto, cualquier variable modificada en el interior de este bucle pierde su valor a la salida!

## c. Bucle until

El comando **until** permite un bucle condicional «hasta». En cuanto se haya cumplido la condición, se sale del bucle.

```
until condición
do
    comandos
done
```

```
until
bloque de instrucciones que forman la condición
do
    comandos
done
```

## d. true y false

El comando **true** no hace más que volver a enviar 0. El comando **false** siempre devuelve 1. De esta manera, es posible realizar bucles sin fin. La única manera de salir de estos bucles es con un exit o un break.

Por convención, cualquier programa que no devuelve error devuelve 0, mientras que cualquier programa que devuelve un error, o un resultado que se debe interpretar, devuelve otra cosa que no es 0. Es lo contrario en lógica booleana.

```
while true
do
    comandos
    exit / break
done
```

## e. break y continue

El comando **break** permite interrumpir un bucle. En este caso, el script continúa su ejecución después del comando **done**. Puede coger un argumento numérico que indique el número de bucles a saltar, en el caso de los bucles anidados (rápidamente ilegible).

El comando **continue** permite saltarse líneas de una iteración e ir directamente a la siguiente. Puede coger

```

while true
do
    echo "¿Cadena? \c"
    read a
    if [ -z "$a" ]
    then
        break
    fi
done

```

un argumento numérico, que indica el número de bucles que se deben iniciar de nuevo (se remontan n bucles). El script se ejecuta con el comando **do**.

## f. Bucle select

El comando **select** permite crear menús simples, con selección por número. La inserción se efectúa por el teclado con el prompt de la variable PS3. Si el valor insertado no es correcto, se efectúa un bucle y se visualiza de nuevo el menú. Para salir de un select, hay que utilizar un **break**.

```

select variable in lista_contenido
do
    tratamiento
done

```

Si no se especifica `in lista_contenido`, se utilizarán y visualizarán los parámetros de posición.

```

$ cat select.sh
#!/bin/bash
PS3="Su elección:"
echo "¿Qué dato?"
select respuesta in Julio Román Francisco sale
do
    if [[ "$respuesta" = "sale" ]]
    then
        break
    fi
    echo "Usted eligió $respuesta"
done
echo "Adiós."
exit 0

$ ./select.sh
¿Qué dato?
1) Julio
2) Román
3) Francisco
4) sale
Su elección :1
Usted eligió Julio
Su elección :2
Usted eligió Román
Su elección :3
Usted eligió Francisco
Su elección :4
Adiós.

```

## 10. Las funciones

Las funciones son trozos de scripts con nombre, directamente llamados por su nombre, que pueden aceptar parámetros y devolver valores. Los nombres de funciones siguen las mismas reglas que las variables, excepto

que no se pueden exportar.

```
nombre_funcion ()  
{  
    comandos  
    return  
}
```

Las funciones pueden escribirse o bien en el script actual, o bien en otro archivo que puede incluirse en el entorno. Para ello, teclee:

```
. nombrefic
```

El punto seguido de un nombre de fichero carga su contenido (funciones y variables) en el contexto actual.

El comando **return** permite asignar un valor de vuelta a una función. Bajo ningún concepto se debe utilizar el comando **exit** para salir de una función porque esta instrucción interrumpe también el script invocante.

```
$ cat funcion  
ll ()  
{  
    ls -l $@  
}  
li ()  
{  
    ls -i $@  
}  
$ . funcion  
$ li  
252 case1.sh      326 for4.sh      187 param.sh      897 resultado.txt  
568 dump.log       409 vinculo_fic1  272 param2.sh    991 pepito.txt  
286 funcion        634 vinculo_fic2  260 param3.sh    716 ver_a.sh  
235 for1.sh        1020 lista       42 param4.sh    1008 while1.sh  
909 for2.sh        667 lista1      304 read.sh      789 for3.sh  
789 for3.sh        1006 lista2      481 dir1
```

## 11. Cálculos y expresiones

### a. expr

El comando **expr** permite efectuar cálculos sobre valores numéricos, comparaciones, así como la búsqueda en cadenas de texto.

Operador	Papel
+	Suma.
-	Sustracción.
*	Multiplicación. Como el shell reconoce la estrella en tanto que comodín, hay que cerrarla con una contrabarra: \*.
/	División.
%	Módulo.

<b>!=</b>	Diferente. Visualiza 1 si diferente, 0 en caso contrario.
<b>=</b>	Igual. Visualiza 1 si igual, 0 en caso contrario.
<b>&lt;</b>	Inferior. Visualiza 1 si inferior, 0 en caso contrario.
<b>&gt;</b>	Superior. Visualiza 1 si superior, 0 en caso contrario.
<b>&lt;=</b>	Inferior o igual. Visualiza 1 si inferior, 0 en caso contrario.
<b>&gt;=</b>	Superior o igual. Visualiza 1 si superior, 0 en caso contrario.
<b>:</b>	Búsqueda en una cadena. P. ej.: expr Julio: J* devuelve 1, ya que Julio empieza por J. Sintaxis particular: expr "Julio": ".*" devuelve la longitud de la cadena.

```
$ expr 7 + 3
10
$ expr 7 \* 3
21
$ a=$(expr 13 - 10)
$ echo $a
3
$ cat expr1.sh
#!/bin/bash
cumul=0
contador=0
nb_bucles=10
while [ "$contador" -le "$nb_bucles" ]
do
    cumul=$(expr $cumul + $contador)
    echo "$cumul=$cumul, bucle=$contador"
    contador=$(expr $contador + 1)
done
$ ./expr1.sh
cumul=0, bucle=0
cumul=1, bucle=1
cumul=3, bucle=2
cumul=6, bucle=3
cumul=10, bucle=4
cumul=15, bucle=5
cumul=21, bucle=6
cumul=28, bucle=7
cumul=36, bucle=8
cumul=45, bucle=9
cumul=55, bucle=10
$ expr "Julio César" : ".*"
11
```

## b. Cálculos con bash

El bash propone una forma sencilla de cálculos sobre los enteros colocando la operación entre **\$((...))**:

```
$ a=1
$ a=$((a+1))
$ echo $a
2
$ b=2
$ a=$((a*b))
```

```
$ echo $a  
4
```

No necesita especificar los \$ de los nombres de las variables entre los dobles paréntesis.

## 12. Punteros

Aquí tiene un ejemplo:

```
$ a=Julio  
$ b=a  
$ echo $b  
a
```

¿Cómo visualizar el valor de a y no su nombre? Utilizando el comando **eval**. Este comando, situado al principio de la línea, intenta interpretar el valor de una variable precedida de dos «\$» como si fuera otra variable.

```
$ eval echo \$\$b  
Julio  
  
$ cat eval.sh  
cpt=1  
for i in a b c  
do  
    eval $i=$cpt  
    cpt=$((cpt+1))  
done  
echo $a $b $c  
  
$ ./eval.sh  
1 2 3
```

## 13. Tratamiento de señales

El comando **trap** permite modificar el comportamiento del script en el momento de la recepción de un señal.

Comando	Reacción
trap " señales	Ignora las señales. trap " 2 3 ignora las señales 2 y 3.
trap 'comandos' señales	Para cada señal recibida, ejecuta los comandos indicados.
trap señales	Restaura las acciones por defecto para estas señales.

En el ejemplo siguiente **trap** impide la ejecución de [Ctrl] C (SIGINT) e intercepta la señal SIGTERM:

```
$ cat trap.sh  
#!/bin/bash  
  
salir()  
{  
    echo "Señal 15 recibido"  
    exit 0  
}  
  
trap '' 2
```

```
trap exit 15

while true
do
    echo "Ctrl+C imposible!"
done

$ ./trap.sh
Ctrl+C imposible!
...
...
```

Desde otra consola:

```
$ kill -2 12456 # ningun efecto
$ kill -15 12456 # SIGTERM
En la primera consola:
Ctrl+C imposible!
Ctrl+C imposible!
Ctrl+C imposible!
Ctrl+C imposible!
Ctrl+C imposible!
Ctrl+C imposible!
Señal 15 recibida
```

## 14. Comando «::»

El comando «::» suele ser totalmente desconocido para los usuarios de Unix. Siempre devuelve el valor 0 (éxito). Por lo tanto, se puede utilizar para sustituir al comando **true** en un bucle, por ejemplo:

```
while:
do
...
done
```

Este comando, colocado al principio de línea y seguido de otro comando, trata el comando y sus argumentos, pero no hace nada y siempre devuelve 0. Puede ser útil para probar variables.

```
$ : ls
$ : ${X:?Error}
X : Error
```

# SQL

## 1. Presentación

El SQL, *Structured Query Language*, es un lenguaje estandarizado ISO de consulta y tratamiento de bases de datos relacionales. Se pueden descomponer sus posibilidades en cuatro funciones:

- El lenguaje de **definición de datos**: creación, modificación y supresión de las tablas, índice, coacciones, etc.
- El lenguaje de **tratamiento de datos**: seguramente el más famoso. Concierne a las peticiones «clásicas»: añadido, supresión, modificación y selección de datos.
- El lenguaje de **control de datos**: instalación y gestión de los privilegios de los usuarios de la base de datos.
- El lenguaje de **control de las transacciones**: gestión en particular de los «commit» y de las posibles vueltas atrás, procedimientos, etc.

Puede ser útil para un administrador saber crear peticiones en una base de datos: algunas herramientas se sirven de bases SQL para almacenar sus datos. Pero incluso en el caso contrario, como se utiliza mucho y es muy práctico, resulta útil conocer SQL. Esta iniciación le muestra cómo efectuar las cuatro peticiones básicas: selección, añadido, modificación y supresión de datos.

El modelo básico de datos utilizado para esta presentación se halla en el anexo de este libro.

## 2. Peticiones de selección

### a. Select

Las selecciones, con la instrucción SELECT, son las principales peticiones utilizadas en SQL, que permiten extraer datos de una o varias tablas según determinados criterios.

```
SELECT apellido_campo1, apellido_campo2...
FROM table;
```

Se devuelve el resultado en forma de tabla cuyos encabezamientos son los nombres de los campos seleccionados. Es posible renombrarlos con AS. Por ejemplo:

```
SELECT apellido, nombre, id AS identificador FROM t_usuarios;
```

La utilización del asterisco «\*» como campo permite seleccionar todos los campos de la tabla. La petición siguiente lista todo el contenido de la tabla:

```
SELECT * FROM t_usuarios;
```

### b. Distinct

Si la petición devuelve líneas idénticas, puede suprimir los duplicados con **DISTINCT**. En el caso siguiente, es lógico pensar que varios usuarios tienen el mismo nombre. La causa DISTINCT va a suprimir en salida los duplicados (o triplicados, etc.).

```
SELECT DISTINCT nombre FROM t_usuarios;
```

## c. Where

La cláusula WHERE especifica las condiciones de selección de las líneas.

```
SELECT apellido_campo1, apellido_campo2...
FROM table
WHERE condición;
```

Es posible aplicar una condición a cualquier campo de la tabla, incluyendo aquellos que no están presentes en la selección de campos. Teniendo en cuenta el tipo de datos del campo o de los campos afectados por la condición, puede utilizar distintos operadores: =, >=, <=, >, <, <> (diferente) y la negación con el signo de exclamación: != (diferente), !> (no superior), !< (no inferior). Se pueden relacionar los criterios de condiciones con los operadores lógicos AND, OR y NOT. Si el criterio es un texto, se pone entre comillas simples: 'titi'.

```
SELECT nombre FROM t_usuarios WHERE apellido='SANCHEZ';
```

### Atributos de Where

Puede utilizar los atributos LIKE, BETWEEN o IN en una cláusula WHERE.

- LIKE compara el valor de un campo con un valor de texto especificado con posibles caracteres de sustitución: «%» para una cadena cualquiera y «\_» para un carácter cualquiera.
- BETWEEN especifica un intervalo de datos.
- IN propone una lista de elementos.

Esta petición extrae todos los registros cuyos id están incluidos entre 0 y 100.

```
SELECT * FROM t_usuarios WHERE id BETWEEN 0 AND 1000;
```

Ésta extrae todos los registros en los que el nombre corresponde a un nombre compuesto «Juan »: Juan Jacobo, Juan Carlos, Juan María, etc.

```
SELECT * FROM t_usuarios WHERE nombre LIKE 'Juan %';
```

Finalmente, ésta última extrae los usuarios que viven en Madrid, Barcelona, Valencia y Sevilla.

```
SELECT * FROM t_usuarios WHERE ciudad IN ('Madrid', 'Barcelona', 'Valencia', 'Sevilla');
```

## 3. Las expresiones y las funciones

Puede utilizar expresiones, en particular las aritméticas, dentro de una instrucción SELECT, tanto en la selección de los campos como en la cláusula WHERE. La petición siguiente calcula el precio NETO de un producto considerando que el campo iva contiene el porcentaje de IVA aplicado al producto.

```
SELECT precio+(precio*iva) AS precio_net FROM t_productos WHERE id_producto='1245';
```

Si los campos son de texto, el símbolo «+» permite hacer concatenaciones:

```
SELECT apellido+' '+nombre FROM t_usuarios;
```

Las funciones se aplican como las expresiones en los campos seleccionados o en la cláusula WHERE. Son de tres tipos principales: matemáticas y estadísticas, fechas y cadenas de caracteres. Le presentamos unos ejemplos:

Esta consulta cuenta el número de líneas en una tabla:

```
SELECT count(*) FROM t_usuarios;
```

Ésta extrae los precios mínimo, máximo y medio del conjunto de la tabla de los productos:

```
SELECT min(precio), max(precio), avg(precio) FROM t_productos;
```

Esta última extrae el importe total de las existencias de productos:

```
SELECT sum(precio*ctd) from t_productos;
```

Las funciones de fecha se aplican sobre los campos de tipo fecha, salvo `current_date`, que devuelve la fecha del día:

```
SELECT current_date;
```

La petición siguiente extrae todos los usuarios que se han inscrito en 2008:

```
SELECT * from t_usuarios WHERE month(f_inscripción)='2008';
```

Finalmente, aquí tiene un ejemplo de petición que extrae todos los usuarios cuyo nombre empieza por un 'RI' al recuperar los dos primeros caracteres del nombre con la función `LEFT`:

```
SELECT * FROM t_usuarios WHERE left(nombre,2) = 'RI';
```

## 4. La cláusula ORDER BY

La cláusula `ORDER BY` permite ordenar los resultados. Es posible indicar el tipo de ordenación, creciente con `ASC`, decreciente con `DESC`. La petición siguiente ordena los productos de más barato a más caro teniendo en cuenta el IVA:

```
SELECT precio+(precio*iva) AS precio_net FROM t_productos ORDER BY precio_net ASC;
```

## 5. La cláusula GROUP BY

La cláusula `GROUP BY` agrupa los resultados por campos de valores idénticos. La petición siguiente visualiza el id de proveedor y número total de productos en reserva por id de proveedor:

```
SELECT id_proveedor, sum(ctd) FROM t_productos GROUP BY id_proveedor;
```

La cláusula `HAVING` se utiliza con `GROUP BY` y permite reducir la selección con un criterio en el campo o los campos de agrupación. Por ejemplo, en la petición de más arriba, puede agrupar únicamente en un solo proveedor dado:

```
SELECT id_proveedor, sum(ctd) FROM t_productos GROUP BY id_proveedor HAVING id_proveedor='1';
```

## 6. Las combinaciones

Hasta ahora las consultas afectaban a una única tabla, pero con SQL es posible recuperar registros de varias tablas al mismo tiempo. Para ello las tablas implicadas deben compartir campos que las puedan relacionar. La combinación de tablas se basa en la relación de varias tablas a través de uno (o varios) campos (atributos) de cada tabla. No hay límite al número de tablas ni al de combinaciones.

```
A  
SELECT tabla1.campo1, tabla2.campo1  
FROM tabla1, tabla2  
WHERE tabla1.campocomun=tabla2.campocomun;
```

continuación presentamos una consulta que extrae el nombre de cada proveedor para cada producto:

```
SELECT DISTINCT t_proveedores.nombre  
FROM t_proveedores, t_productos  
WHERE t_proveedores.id_proveedor=t_proveedores.id_proveedor;
```

Se puede utilizar el AS para renombrar las tablas en la salida de la consulta. Si el nombre del campo es único en todas las tablas, entonces no hace falta especificar la tabla. Se puede transformar la consulta de la manera siguiente:

```
SELECT DISTINCT nombre  
FROM t_proveedores as t1, t_productos as t2  
WHERE t2.id_proveedor=t1.id_proveedor
```

Se pueden efectuar combinaciones con más de dos tablas. Así, para obtener la lista de los proveedores por clientes, necesitará consultar la tabla de los clientes llamada "t\_usuarios", la tabla de los pedidos, la tabla de los productos y la tabla de los proveedores:

```
SELECT DISTINCT t_proveedores.nombre  
FROM t_usuarios, t_pedidos, t_productos  
WHERE t_usuarios.id=t_pedidos.id  
AND t_pedidos.id_producto=t_productos.id_producto  
AND t_productos.id_proveedor=t_proveedores.id_proveedor  
AND t_usuarios.id='1'
```

Es posible combinar una tabla consigo misma. El truco consiste en duplicar temporalmente la tabla después de FROM, pero con nombres diferentes especificados con AS. El ejemplo siguiente extrae los apellidos y nombres de alias del usuario 2:

```
SELECT B.apellido, B.nombre  
FROM t_usuarios AS A, t_usuarios AS B  
WHERE B.id = A.id_alias  
AND A.id = '2';
```

## 7. Subconsultas

Es posible hacer selecciones anidadas: que la devolución de una primera selección sirve de criterio para una segunda selección. Por ejemplo, veamos cómo extraer los apellidos y nombres de todos los alias:

```
Si  
la  
SELECT id  
FROM t_usuarios  
WHERE id IN (select id_alias from t_usuarios WHERE id_alias!='')
```

subconsulta devuelve un único valor, puede usar un igual para pasarle el valor a la consulta superior. En caso de varios valores devueltos, utilice IN.

## 8. Las inserciones

Inserte registros en una tabla SQL con el comando INSERT.

```
INSERT INTO table (campo1, campo2, etc.)
VALUES ('valor1', 'valor2'...);
```

Por ejemplo, veamos cómo insertar un nuevo cliente en la tabla t\_usuarios:

```
INSERT INTO t_usuarios (id, apellido, nombre, ciudad, id_alias)
VALUES ('3', 'Aguilera', 'Julio', 'Olmo', NULL);
```

## 9. Actualizaciones

La instrucción UPDATE actualiza uno o varios registros de una tabla. La cláusula WHERE es opcional. En el ejemplo, se actualizan todos los registros.

```
UPDATE tabla SET campol='valor', campo2='valor'
[WHERE campon='valor']
```

Veamos cómo aumentar todos los precios un 5%:

```
UPDATE t_productos SET precio=precio*1.05;
```

## 10. Supresión

La instrucción DELETE suprime uno o varios registros de una o varias tablas. Cuidado, según el modelo básico de datos, con las restricciones de integridad: en algunos casos, suprimir una referencia puede suprimir n registros en cascada.

```
DELETE FROM tabla
WHERE campol='valor';
```

Veamos cómo suprimir el usuario 3 creado más arriba:

```
DELETE from t_usuarios WHERE id=3;
```

# Validación de los conocimientos adquiridos: preguntas-respuestas

## 1. Preguntas

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### El shell

**1** ¿Qué es un shell (más de una respuesta puede ser correcta)?

- A - Un intérprete de comandos.
- B - Un lenguaje de programación.
- C - Una herramienta de gestión de ficheros.
- D - Una consola en el interior de una interfaz gráfica.

**2** ¿Cuál es el shell por defecto bajo Linux?

- A - ksh
- B - sh
- C - bash
- D - tcsh

**3** Supongamos la línea siguiente: pepito@eeepc:~/Documentos/Fotos>, ¿qué información se puede obtener?

**4** La línea siguiente echo "hola, la fecha es " ; date ; exit:

- A - Visualiza el texto y luego la fecha y espera una inserción.
- B - No hace nada: usted sale.
- C - Sale únicamente si el comando **date** funciona.
- D - Como A, pero luego deja el shell.

**5** Ha insertado un comando que no se detiene (p. ej.: yes). ¿Qué hacer para terminar el programa?

### Historial

**6** ¿Cómo volver a llamar al último comando insertado con el teclado?

**7** Misma pregunta, pero con el comando **fc**.

### Gestión de ficheros

**8** ¿Es un programa ejecutable un fichero especial?

**9** ¿Cuál es la longitud máxima de un nombre de fichero, extensión incluida?

- A - 255 caracteres.
- B - 32 caracteres.
- C - 255 caracteres más la extensión.
- D - La pregunta es errónea.

**10** ¿Cuáles de las siguientes rutas son rutas absolutas?

- A - ~/Documentos

- B - /usr/local/bin
- C - /opt/kde3/bin/../lib
- D - /home/seb/Documentos

**11** Esté donde esté, ¿cuál es el efecto de un «cd» sin parámetros?

**12** Para listar de manera recursiva todos los ficheros sin excepción presentes en /usr, ¿qué comando se debe teclear?

**13** En un directorio que contiene una multitud de ficheros, ¿cómo visualizar la lista detallada de los ficheros, pero de modo que los últimos modificados aparezcan en último lugar en la lista (todo en la parte inferior)?

**14** Quiere crear, de la forma más sencilla posible, la ruta fuentes/C/backup en su directorio personal. Sabiendo que ninguno de estos directorios existe, inserta:

- A - mkdir fuentes/C/backup
- B - mkdir -p fuentes/C/backup
- C - mkdir fuentes ; cd fuentes ; mkdir C ; cd C; mkdir backup
- D - mkdir -p fuentes C backup

**15** Quiere suprimir el directorio C de arriba de la manera siguiente: rmdir C. ¿Va a funcionar?

**16** Como root, tiene que mover un directorio y su contenido, que pertenecen al usuario «oracle», a otro lugar. ¿Cuáles son los parámetros de cp que va a utilizar?

**17** ¿Cuál es el efecto, como root, del comando **rm -rf /\*?**

**18** ¿Cuál es el efecto, como simple usuario, del comando **rm -rf /\*?**

- A - Se destruye todo el sistema de ficheros.
- B - Ningún efecto: no tengo derechos para suprimir la raíz.
- C - Se devuelve un error, ya que sólo root puede utilizar este comando.
- D - Se suprimen mis ficheros y directorios, estén donde estén, incluso mi directorio home.

**19** ¿Cómo crear un vínculo simbólico de /tmp/seb hacia /home/seb/tmp?

**20** Mi directorio actual contiene tres ficheros: fic1, fic2, f3. ¿Qué devuelve el comando **ls f\*?[13]**?

## **Búsqueda de ficheros**

**21** Es habitual encontrar ficheros core cuando los programas se cuelgan. ¿Cómo suprimirlos todos?

**22** Quiere listar todos sus ficheros avi de más de 650 MB ubicados en /public/videos y que no ha visto desde hace unos 6 meses. ¿Cómo conseguirlo?

**23** Quiere buscar todas las instancias de un programa llamado «startappli» dentro del path. ¿Qué comando utiliza?

- A - whereis -b startappli
- B - which startappli
- C - which -a startappli
- D - locate startappli

**24** Ha perdido un documento que empieza por una letra, en alguna parte del sistema de ficheros. ¿Cómo comprobar que locate devolverá una información actualizada y cómo obtener la lista de los ficheros que podrían corresponder al que busca?

## **El editor vi**

**25** ¿Cómo entrar en modo inserción para insertar algo de texto en el sitio actual, sea cual sea la palabra

anterior?

- A - [Esc] i
- B - [Esc] a
- C - [Esc]: i
- D - : i

**26** ¿Cómo guardar su documento actual como «newfic»?

**27** ¿Cuáles son los tres medios de dejar vi guardando su fichero?

- A - ZZ
- B - :q
- C - :wq
- D - : x

**28** ¿Cómo buscar todas las líneas que empiezan con hola u Hola?

**29** Quiere numerar las líneas. ¿Cómo conseguirlo?

**30** ¿Cómo dirigirse a la línea 112, copiar 8 líneas y ubicarlas después de la línea 180?

### **Las redirecciones**

**31** ¿Cuál es el resultado del comando >fic sin nada delante?

**32** ¿Qué va a mostrar ls -R >lista y qué contendrá lista si usted ejecuta este comando como usuario normal?

**33** Como usuario, para listar la totalidad de los ficheros del sistema y colocar todos los resultados, sean cuales sean, en una lista, ¿qué redirección debe indicar?

- A - 2>dev/null >lista
- B - >lista 2>lista
- C - 1>lista 2>&1
- D - 2>1 >lista

**34** ¿Cómo pasar un mensaje mediante echo por el canal de error?

**35** ¿Cómo visualizar el contenido de un fichero con cat y una redirección en entrada?

**36** ¿Cómo suprimir todos los mensajes de error?

- A - 2>/dev/null
- B - 2>/dev/zero
- C - >/dev/null 2>&1
- D - 2>fic ; rm fic

### **Filtros y herramientas**

**37** Busque en /etc/passwd la línea correspondiente a su usuario, y únicamente a éste.

**38** Busque en /etc/passwd los usuarios que no tienen bash como shell por defecto. Utilice una tubería (pipe).

**39** Usando una simple expresión regular, suprima las líneas vacías y los comentarios (que empiezan por #) de /etc/hosts.

**40** Visualice únicamente la lista de los logins y UID asociados (campo 3) de /etc/passwd.

- 41** ¿Qué serie de comandos permite conocer el número de líneas real (no vacías, y no los comentarios) de /etc/hosts? Utilice dos veces las tuberías y numere las líneas.
- 42** Seleccione el fichero /etc/group por orden de GID (3<sup>a</sup> columna) decreciente.
- 43** Dé la lista de los diferentes shells utilizados por los usuarios declarados en su máquina. No debe haber duplicados.
- 44** En un fichero de texto de su elección (llamado aquí fic) sustituya todas las é, è, ê, ë por una e (cuidado: esta pregunta no funciona en unicode).
- 45** ¿Cómo recuperar la línea 27<sup>a</sup> de un fichero fic (partiendo de la idea de que fic contiene más de 27 líneas)?

- A - cat -n fic | grep ^27 | cut -d" " -f2-
- B - grep -n "" fic | grep ^27 | cut -d: -f2-
- C - tail -27 fic | head -1
- D - head -27 fic | tail -1

**46** Retome la pregunta 33, pero esta vez haga que toda la lista esté ubicada y se visualice en un fichero.

### **Los procesos**

- 47** ¿Es pertinente el comando `tail -f fic &`?
- 48** Ha ejecutado un comando que monopoliza la consola sin que muestre nada, ya que el tratamiento es muy largo. ¿Cómo pasarlo al segundo plano?
- 49** Liste los procesos que pertenecen a root.
- 50** ¿Cómo visualizar la estructura en árbol de todos los procesos? El acceso a la ayuda interna de pspuede ayudarle.
- 51** ¿Cómo saber de una manera muy simple si el proceso firefox está en ejecución?
- 52** Aísle los números de los procesos que corresponden a los diversos shell bash lanzados.
- 53** Ejecute un comando `sleep 100&`. Luego termine correctamente este proceso.
- 54** ¿Qué ocurre si, como root, usted «mata» de manera abrupta el proceso 1?
- 55** Un proceso recalcitrante se niega a pararse. ¿Cuál es la única acción posible?
- 56** ¿Cómo estar seguro de que un proceso importante no se parará cuando usted cierra su consola o terminal?
- 57** Un proceso de codificación llamado transcode consume todos los recursos CPU de su máquina, poniendo a prueba su procesador y haciendo «currar» a su ordenador. ¿Cómo conocer su prioridad actual? ¿Cómo rebajar la prioridad?

### **Ejecución condicional**

- 58** Suprima un fichero fic. Si la supresión tiene éxito, haga que se muestre OK. En caso contrario, haga que se muestre ERROR y haga un `ls -l` de este fichero.

### **Variables**

- 59** Declare una variable VAR que contenga su apellido.
- 60** Una variable a contiene el valor 2. Inicie desde la misma consola un segundo bash. Visualice a. ¿Qué contiene la variable a?
- 61** Cree una variable b que contenga 3 y expórtela. Inicie un bash hijo e incremente b. Salga y muestre b. ¿Qué contiene?
- A - Nada.

- B - 4.
- C - 3 ya que se incrementó b en un proceso hijo diferente del padre.
- D - 1 ya que ya no existe b en los hijos, se ha vuelto a poner a 0.

**62** Al contener texto la variable C, ¿qué muestra echo \${C:+“cucú”} y qué contiene C?

**63** Modifique el path por defecto añadiéndole /opt/bin en primera posición.

**64** ¿Cómo conocer el valor de devolución de una supresión de fichero?

**65** Mande la señal 15 al último proceso ejecutado en segundo plano.

**66** Dé dos medios de sumar dos variables A y B que contienen enteros y de ubicar el resultado en A.

### **Programación del shell**

**67** ¿Por qué es importante empezar un script indicando el tipo de shell?

**68** Se transmiten 10 parámetros a un script. ¿Cómo recuperar el décimo lo más rápido posible?

- A - por \$10.
- B - escribiendo nueve «shift» y recuperando \$1.
- C - un shift y luego \$9.
- D - por: A=10 y luego \${\$A}.

**69** Desde su shell ¿cómo puede recuperar el valor devuelto por «exit» de un script cualquiera?

**70** Conociendo un login ¿cómo recuperar su UID en una variable del mismo nombre?

**71** ¿Qué línea escribir en un script para salir con un mensaje de error y un código de retorno 1 si el número de parámetros pasados es igual a 0?

- A - test \$# != 0 || echo Error ; exit 1
- B - [ \$# -eq 0 ] && { echo Error ; exit 1 ; }
- C - test \$0 -ne 0 && ( echo Error; exit 1)
- D - [[ \$# -eq 0 ]] && echo Error && exit 1

**72** Si el fichero fic existe y es legible, liste su contenido. En caso de que no existe, créelo.

**73** Con la ayuda de un if, si no hay ningún parámetro transmitido al script, ponga dos preposicionados:-l y -r.

**74** Un programa espera tres valores desde el teclado, uno tras otro, separados por espacios, en la misma línea. ¿Qué comando se debe utilizar?

**75** ¿Cómo realizar un bucle para listar todos los parámetros? Elija dos respuestas.

- A - for param ; do echo \$param ; done
- B - while [ \$# -ne 0 ]; do echo \$1; shift ; done
- C - until [ \$# -ne 0 ]; do shift; echo \$1; done
- D - a=1; while [ \$a -ne \$# ]; do echo \${\$a}; a=\$((a+1)); done

**76** ¿Cómo salir de un bucle sin fin?

**77** ¿Cómo programar una función abs que devuelve el valor absoluto de un número?

**78** ¿Cómo impedir la intercepción de la señal 15 en un script?

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /78

Para este capítulo, su resultado mínimo debe ser de 58 respuestas acertadas (58/78).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- El shell.
- Historial.
- Gestión de ficheros.
- Búsqueda de ficheros.
- El editor vi.
- Las redirecciones.
- Filtros y herramientas.
- Los procesos.
- Ejecución condicional.
- Variables.
- Programación de shell.

### 3. Respuestas

#### **El shell**

**1** ¿Qué es un shell (más de una respuesta puede ser correcta)?

- A - Un intérprete de comandos.
- B - Un lenguaje de programación.
- C - Una herramienta de gestión de ficheros.
- D - Una consola en el interior de una interfaz gráfica.

*A y B. El shell permite acceder a todos los comandos del sistema y dispone también de un lenguaje de programación de scripts.*

**2** ¿Cuál es el shell por defecto bajo Linux?

- A - ksh
- B - sh
- C - bash
- D - tcsh

*C, el bash, pero todos están disponibles.*

**3** Supongamos la línea siguiente: pepito@eeepc:~/Documentos/Fotos>, ¿qué información se puede obtener?

*El usuario pepito está conectado a la máquina eeepc. Actualmente está ubicado en la carpeta Documentos/Fotos de su directorio personal. <>> indica que no tiene permisos especiales.*

**4** La línea siguiente echo "hola, la fecha es " ; date ; exit:

- A - Visualiza el texto y luego la fecha y espera una inserción.
- B - No hace nada: usted sale.
- C - Sale únicamente si el comando **date** funciona.
- D - Como A pero luego deja el shell.

*D. Se encadenan los comandos con un punto y coma. Se ejecutan todos en el orden indicado.*

**5** Ha insertado un comando que no se detiene (p. ej.: yes). ¿Qué hacer para terminar el programa?

*Teclee [Ctrl] C.*

### **Historial**

**6** ¿Cómo volver a llamar al último comando insertado con el teclado?

*Con la flecha hacia arriba de las teclas de dirección.*

**7** Misma pregunta, pero con el comando **fc**.

*fc -l -1 (- letra L y - cifra uno).*

### **Gestión de ficheros**

**8** ¿Es un programa ejecutable un fichero especial?

*No, es un fichero ordinario que contiene el código binario del programa con un permiso de ejecución.*

**9** ¿Cuál es la longitud máxima de un nombre de fichero, extensión incluida?

- A - 255 caracteres.
- B - 32 caracteres.
- C - 255 caracteres más la extensión.
- D - La pregunta es errónea.

*D. No hay extensión gestionada a nivel del sistema de ficheros bajo Linux. La extensión es un sufijo incluido en la longitud total del nombre del fichero, de 255 caracteres como máximo.*

**10** ¿Cuáles de las siguientes rutas son rutas absolutas?

- A - ~/Documentos
- B - /usr/local/bin
- C - /opt/kde3/bin/../lib
- D - /home/seb/Documentos

*B y D. C contiene un .. relativo a bin y A es relativo al directorio personal.*

**11** Esté donde esté, ¿cuál es el efecto de un «cd» sin parámetros?

*Le manda automáticamente a su directorio personal.*

**12** Para listar de manera recursiva todos los ficheros sin excepción presentes en /usr, ¿qué comando se debe teclear?

`ls -Ra /usr. R: recursivo, a: all.`

- 13** En un directorio que contiene una multitud de ficheros, ¿cómo visualizar la lista detallada de los ficheros, pero de modo que los últimos modificados aparezcan en último lugar en la lista (todo en la parte inferior)?

`ls -ltr. l lista detallada, t: ordenada por fecha de modificación, r: orden inverso.`

- 14** Quiere crear, de la forma más sencilla posible, la ruta fuentes/C/backup en su directorio personal. Sabiendo que ninguno de estos directorios existe, inserta:

- A - `mkdir fuentes/C/backup`
- B - `mkdir -p fuentes/C/backup`
- C - `mkdir fuentes ; cd fuentes ; mkdir C ; cd C; mkdir backup`
- D - `mkdir -p fuentes C backup`

*B. -p (padre) crea los directorios padres si no existen.*

- 15** Quiere suprimir el directorio C de arriba de la manera siguiente: `rmdir C`. ¿Va a funcionar?

*No. C contiene backup, por lo que no está vacío.*

- 16** Como root, tiene que mover un directorio y su contenido, que pertenecen al usuario «oracle», a otro lugar. ¿Cuáles son los parámetros de `cp` que va a utilizar?

`cp -rp. -r para recursivo, y -p para la preservación de los derechos.`

- 17** ¿Cuál es el efecto, como root, del comando `rm -rf /*?`

*Con este comando, acaba de destruir todo el contenido del sistema de ficheros. Puede presentar su dimisión, si no le han echado antes!*

- 18** ¿Cuál es el efecto, como simple usuario, del comando `rm -rf /*?`

- A - Se destruye todo el sistema de ficheros.
- B - Ningún efecto: no tengo derechos para suprimir la raíz.
- C - Se devuelve un error, ya que sólo root puede utilizar este comando.
- D - Se suprimen mis ficheros y directorios, estén donde estén, incluso mi directorio home.

*ID! El comando es recursivo: busca entrar en todos los directorios, incluso los tuyos.*

- 19** ¿Cómo crear un vínculo simbólico de `/tmp/seb` hacia `/home/seb/tmp`?

`ln -s /tmp/seb /home/seb/tmp`

- 20** Mi directorio actual contiene tres ficheros: `fic1, fic2, f3`. ¿Qué devuelve el comando `ls f*?[13]`?

*fic1 solamente, ya que el fichero debe empezar con una f, luego incluir al menos un carácter y terminar con 1 o 3. fic1 es el único que se corresponde.*

## **Búsqueda de ficheros**

- 21** Es habitual encontrar ficheros core cuando los programas se cuelgan. ¿Cómo suprimirlos todos?

`find / -name "core" -exec rm -f {} \;`

- 22** Quiere listar todos sus ficheros avi de más de 650 MB ubicados en `/public/videos` y que no ha visto desde hace unos 6 meses. ¿Cómo conseguirlo?

```
find /public -user <login> -iname "*.avi" -type f -size +665600k -atime +180 -ls
```

**23** Quiere buscar todas las instancias de un programa llamado «startappli» dentro del path. ¿Qué comando utiliza?

- A - whereis -b startappli
- B - which startappli
- C - which -a startappli
- D - locate startappli

C. -a busca todas las coincidencias con el nombre dado en el path.

**24** Ha perdido un documento que empieza por una letra, en alguna parte del sistema de fichero. ¿Cómo comprobar que locate devolverá una información actualizada y cómo obtener la lista de los ficheros que podrían corresponder al que busca?

Verifique la fecha de modificación del fichero /var/lib/locatedb. Debe ser reciente. Para buscar las coincidencias posibles, utilice locate letra.

### El editor vi

**25** ¿Cómo entrar en modo de inserción para insertar algo de texto en el sitio actual, sea cual sea la palabra anterior?

- A - [Esc] i
- B - [Esc] a
- C - [Esc]: i
- D - : i

A. [Esc] vuelve al modo comando pase lo que pase. Luego basta con pasar al modo inserción con i.

**26** ¿Cómo guardar su documento actual como «newfic»?

Pase al modo comando y guarde con un nuevo nombre, así: [Esc]:w newfic.

**27** ¿Cuáles son los tres medios de dejar vi guardando su fichero?

- A - ZZ
- B - :q
- C - :wq
- D - : x

A, C y D.

**28** ¿Cómo buscar todas las líneas que empiezan con hola u Hola?

:/^hHola

**29** Quiere numerar las líneas. ¿Cómo conseguirlo?

:set num

**30** ¿Cómo dirigirse a la línea 112, copiar 8 líneas y ubicarlas después de la línea 180?

Vaya a la línea 112 :112, teclee `8YY`, desplácese hasta la línea 180 :180 y teclee `p`.

## **Las redirecciones**

**31** ¿Cuál es el resultado del comando `>fic` sin nada delante?

*Esto vacía fic o crea un fichero fic vacío.*

**32** ¿Qué va a mostrar `ls -R >lista` y qué contendrá lista si usted ejecuta este comando como usuario normal?

*Lista va a contener la lista detallada de todos los ficheros y directorios a los cuales el usuario tiene el derecho de acceder, mientras que se visualizarán los errores en la consola: todas las autorizaciones de acceso denegadas por falta de derechos del usuario.*

**33** Como usuario, para listar la totalidad de los ficheros del sistema y colocar todos los resultados, sean cual sean, en una lista, ¿qué redirección debe indicar?

- A - 2>dev/null >lista
- B - >lista 2>lista
- C - 1>lista 2>&1
- D - 2>1 >lista

*C. Primero redirecciona el canal de error en el canal de salida estándar, luego la salida estándar en lista, de derecha a izquierda; el 1 es opcional.*

**34** ¿Cómo pasar un mensaje mediante echo por el canal de error?

*Redireccionando el canal de salida estándar en el canal de error estándar: echo xxx >&2*

**35** ¿Cómo visualizar el contenido de un fichero con cat y una redirección en entrada?

`cat < fichero`

**36** ¿Cómo suprimir todos los mensajes de error?

- A - 2>/dev/null
- B - 2>/dev/zero
- C - >/dev/null 2>&1
- D - 2>fic ; rm fic

*A. Redireccionne el canal de error estándar a /dev/null.*

## **Filtros y herramientas**

**37** Busque en /etc/passwd la línea correspondiente a su usuario, y únicamente a éste.

`grep "login:" /etc/passwd. Cada línea empieza por el login.`

**38** Busque en /etc/passwd los usuarios que no tienen bash como shell por defecto. Utilice una tubería (pipe).

`cat /etc/passwd | grep -v "bash$". El shell es la última columna, elimine todo lo que termina por bash.`

**39** Usando una simple expresión regular, suprima las líneas vacías y los comentarios (que empiezan por #) de /etc/hosts.

`grep -v -E "^(#| $)" /etc/hosts. Una línea vacía comienza por ^ y termina por $.`

**40** Visualice únicamente la lista de los logins y UID asociados (campo 3) de /etc/passwd.

```
cut -d":" -f1,3 /etc/passwd.
```

**41** ¿Qué serie de comandos permite conocer el número de líneas real (no vacías, y no los comentarios) de /etc/hosts? Utilice dos veces las tuberías y numere las líneas.

```
cat -n /etc/hosts | grep -v -E "^(#| $)" | wc -l
```

**42** Seleccione el fichero /etc/group por orden de GID (3<sup>a</sup> columna) decreciente.

```
sort -n -r -t: -k 3 /etc/group
```

**43** Dé la lista de los diferentes shells utilizados por los usuarios declarados en su máquina. No debe haber duplicados.

```
cut -d: -f7 /etc/passwd | sort | uniq
```

**44** En un fichero de texto de su elección (llamado aquí fic) sustituya todas las é, è, ê, ë por una e (cuidado: esta pregunta no funciona en unicode).

```
cat fic | tr "éèêë" "e"
```

**45** ¿Cómo recuperar la línea 27<sup>a</sup> de un fichero fic (partiendo de la idea de que fic contiene más de 27 líneas)?

- A - cat -n fic | grep ^27 | cut -d" " -f2-
- B - grep -n "" fic | grep ^27 | cut -d: -f2-
- C - tail -27 fic | head -1
- D - head -27 fic | tail -1

*D. Las respuestas A y B en el estado actual ya no funcionan si se llega a 270 o 2700, etc.*

**46** Retome la pregunta 33, pero esta vez haga que toda la lista esté ubicada y se visualice en un fichero.

*Utilizando tee: 2>&1 | tee liste*

### **Los procesos**

**47** ¿Es pertinente el comando tail -f fic &?

*No, el comando sigue efectuando su salida. Tratar de ponerlo en segundo plano es inútil.*

**48** Ha ejecutado un comando que monopoliza la consola sin que muestre nada, ya que el tratamiento es muy largo. ¿Cómo pasarlo a segundo plano?

*Suspenda el programa con [Ctrl] Z; luego teclee bg. El programa pasa a segundo plano. Retómelo si es preciso con el comando fg.*

**49** Liste los procesos que pertenecen a root.

```
ps -u root
```

**50** ¿Cómo visualizar la estructura en árbol de todos los procesos? El acceso a la ayuda interna de ps puede ayudarle.

*Visualice la ayuda de ps con ps --help. Tiene dos soluciones: utilice -H, que da la jerarquía, y f, que hace lo mismo pero con un árbol, o ps -eH o ps -e f.*

**51** ¿Cómo saber de una manera muy simple si el proceso firefox está en ejecución?

```
ps -e|grep firefox
```

**52** Aísle los números de los procesos que corresponden a los diversos shell bash lanzados.

*ps -e|grep bash| cut -c1-5. La primera columna se extiende siempre en 5 caracteres.*

**53** Ejecute un comando sleep 100&. Luego termine correctamente este proceso.

*Apunte el número de PID del proceso iniciado en segundo plano, luego efectúe un kill -15 del PID apuntado.*

**54** ¿Qué ocurre si, como root, usted «mata» de manera abrupta el proceso 1?

*Se trata de init. Si usted lo «mata», se van a suprimir todos los procesos, incluso init. Sólo tendrá que reiniciar.*

**55** Un proceso recalcitrante se niega a pararse. ¿Cuál es la única acción posible?

*Un kill -9 debería descartarlo. Sin embargo, hay dos casos en los cuales es imposible, en particular si el proceso está en espera de una entrada-salida, en el disco por ejemplo. En este caso, un reboot puede ser su única opción.*

**56** ¿Cómo estar seguro de que un proceso importante no se parará cuando usted cierra su consola o terminal?

*Hay que iniciarla con nohup y en segundo plano al mismo tiempo.*

**57** Un proceso de codificación llamado transcode consume todos los recursos CPU de su máquina, poniendo a prueba su procesador y haciendo «currar» a su ordenador. ¿Cómo conocer su prioridad actual? ¿Cómo rebajar la prioridad?

*La columna NI visible con el -I de ps le facilita el factor de incremento del scheduler. A menudo está a 0 (prioridad máxima por defecto para el usuario). Rebaje la prioridad del proceso con el comando renice: renice 10 PID.*

## **Ejecución condicional**

**58** Suprime un fichero fic. Si la supresión tiene éxito, haga que se muestre OK. En caso contrario, haga que se muestre ERROR y haga un ls -l de este fichero.

```
rm -f fic && echo "OK" || ( echo "ERROR" ; ls -l fic)
```

## **VARIABLES**

**59** Declare una variable VAR que contenga su apellido.

```
VAR=apellido
```

**60** Una variable a contiene el valor 2. Inicie desde la misma consola un segundo bash. Visualice a. ¿Qué contiene la variable a?

*Nada. Se ha olvidado de exportarlo.*

**61** Cree una variable b que contenga 3 y expórtela. Inicie un bash hijo e incremente b. Salga y muestre b. ¿Qué contiene?

- A - Nada.
- B - 4.
- C - 3 ya que se incrementó b en un proceso hijo diferente del padre.
- D - 1 ya que ya no existe b en los hijos, se ha vuelto a poner a 0.

C. El hijo recupera una copia de b y lo modifica. El padre guarda su propia variable b no modificada.

**62** Al contener texto la variable C, ¿qué muestra echo \${C:+“cucú”} y qué contiene C?

Muestra «cucú», pero C conserva su valor original.

**63** Modifique el path por defecto añadiéndole /opt/bin en primera posición.

```
export PATH=/opt/bin:$PATH
```

**64** ¿Cómo conocer el valor de devolución de una supresión de fichero?

```
rm fic ; echo $?
```

**65** Mande la señal 15 al último proceso ejecutado en segundo plano.

```
kill -15 $!
```

**66** Dé dos medios de sumar dos variables A y B que contienen enteros y de ubicar el resultado en A.

1<sup>a</sup> posibilidad: A=\$((A+B)). 2<sup>a</sup> posibilidad: typeset -i A ; typeset -i B ; A=A+B.

### **Programación de shell**

**67** ¿Por qué es importante empezar un script indicando el tipo de shell?

Porque el shell corriente lee esta primera línea e inicia el shell correspondiente para ejecutar el script. En caso contrario, el shell considera que el lenguaje es el suyo, incluso si el script contiene algo de PHP o de Perl.

**68** Se transmiten 10 parámetros a un script. ¿Cómo recuperar el décimo lo más rápido posible?

- A - por \$10.
- B - escribiendo nueve «shift» y recuperando \$1.
- C - un shift y luego \$9.
- D - por: A=10 y luego \${\$A}.

C. Un shift desplaza todos los argumentos de una unidad, el décimo parámetro se convierte en el noveno; por lo tanto, \$9.

**69** Desde su shell ¿cómo puede recuperar el valor devuelto por «exit» de un script cualquiera?

Se trata del código de retorno del comando, recuperado por \$?.

**70** Conociendo un login ¿cómo recuperar su UID en una variable del mismo nombre?

Por una sustitución de comando: UID=\$(grep ^login /etc/passwd | cut -d: -f3)

**71** ¿Qué línea escribir en un script para salir con un mensaje de error y un código de retorno 1 si el número de parámetros pasados es igual a 0?

- A - test \$# != 0 || echo Error ; exit 1
- B - [ \$# -eq 0 ] && { echo Error ; exit 1 ; }
- C - test \$0 -ne 0 && ( echo Error; exit 1)
- D - [[ \$# -eq 0 ]] && echo Error && exit 1

B y D son correctos.

**72** Si el fichero fic existe y es legible, liste su contenido. En caso de que no exista, créelo.

```
[[ -e fic -a -r fic ]] && cat fic || touch fic. O también [ -r fic
```

# Prácticas

## 1. Gestión de los ficheros

Objetivo: efectuar operaciones básicas en el sistema de ficheros.

- A partir de su directorio personal, cree la estructura siguiente utilizando un único comando:

```
|-----carpeta1  
|       |-----carpeta3  
|-----carpeta2  
|       |-----carpeta4
```

Utilice el comando **mkdir** con el parámetro **-p**:

```
$ mkdir -p carpeta1/carpeta3 carpeta2/carpeta4
```

- Vaya al directorio carpeta1 con una ruta absoluta y cree el fichero fichero1 en este directorio. La ruta absoluta sale de la raíz sin ninguna ruta relativa. Teclee:

```
$ cd /home/user/carpeta1
```

Cree el fichero con touch:

```
$ touch fichero1
```

- Copie fichero1 en el directorio carpeta3 con una ruta relativa. La ruta es relativa en función de la ubicación actual: .. sube un nivel, y . define la ubicación corriente.

```
$ pwd  
/home/user/carpeta1
```

Copie simplemente el fichero en la carpeta3 donde está usted ahora:

```
$ cp fichero1 carpeta3
```

o también:

```
$ cp fichero1 ./carpeta3
```

- Vaya a la carpeta2 utilizando una ruta relativa, y copie el fichero fichero1 de carpeta3 con el nombre fichero2 donde se encuentra usted. Para moverse:

```
$ cd ../carpeta2
```

Para copiar el fichero:

```
$ cp ../carpeta1/carpeta3 ./fichero2
```

- Renombre fichero2 como fichero3 y muévalo al directorio carpeta3. El comando **mv** mueve y renombra.

```
$ mv fichero2 ../carpeta1/carpeta3/fichero3
```

6. Suprima fichero1 del directorio carpeta3.  
El comando **rm** suprime el fichero.

```
$ rm ./carpeta1/carpeta3/fichero1
```

7. Con rmdir suprime carpeta2, luego carpeta1 y todo su contenido. ¿Es posible? ¿Por qué?  
¿Cómo conseguirlo?  
No puede suprimir carpeta2 directamente con rmdir, ya que contiene carpeta4 y por lo tanto no está vacío. Tiene que pasar por el comando **rm** con el parámetro **-r**.

```
$ cd  
$ rm -rf carpeta2
```

8. ¿Cuál es el objetivo del comando `ls -l [a-z]*. ??[ !0-9]`?  
El fichero empieza con una letra de a a z y termina por cuatro caracteres: el punto, dos caracteres cualesquiera y el último, que es cualquier cosa excepto una cifra. Se muestran los detalles de los ficheros que corresponden a este criterio.

9. Cree un fichero llamado «-i» con una redirección: echo > -i. Intente suprimirlo.  
Es un clásico. Si usted intenta suprimir el fichero, rm devuelve un error que indica que falta un parámetro (el nombre del fichero). Interpreta -i como una opción del comando. Así, la pregunta es: ¿cómo conseguir que -i no sea reconocido como una opción?  
Como se considera todo lo que empieza por un guión como un parámetro, debe indicar una ruta que permita aislar el guión:

```
$ rm ./-i
```

## 2. Buscar ficheros

Objetivo: buscar ficheros con find, whereis y locate.

1. Visualice todos los ficheros que tienen un tamaño inferior a 400 bytes y los derechos 644. Utilice los parámetros **-size** y **-perm** del comando **find**:

```
$ find / -size -400c -perm 644 -print
```

2. Visualice todos los ficheros en su directorio personal que tienen un tamaño inferior a 400 bloques.

```
$ find ~ -size -400 -print
```

3. Liste en formato largo todos los ficheros de su propio sistema que fueron modificados hace más de 7 días.

Utilice los parámetros **-user** y **-mtime**:

```
$ find / -user seb -mtime +7 -ls
```

4. Liste y visualice en formato largo los ficheros en su directorio personal que tienen como propietario guest o que tienen un tamaño entre 512 y 1024 bytes, ambas cantidades incluidas.

La pequeña trampa reside aquí en el «incluidas». Si usted indica +512c, se excluyen los ficheros de 512 bytes. Debe modificar los límites en consecuencia.

```
$ find ~ -user guest -size +511c -size -1025c -ls
```

5. Busque todos los ficheros vacíos del sistema que no pertenecen a root e intente suprimirlos.

Utilice los parámetros `-empty` y `-exec` para ejecutar un `rm` en cada fichero encontrado.

```
$ find / -type f -empty -exec rm -f {} \;
```

6. Indique dónde se encuentra el comando binario `ls`. Utilice el comando **whereis** para ello:

```
$ whereis -b ls
```

7. Ha perdido el fichero `letra_importante.odf`. ¿Dónde está, sin utilizar `find`? Para contestar hace falta que la base `locatedb` esté construida con `updatedb`. Luego, utilice el comando **locate**:

```
$ locate letra_importante.odf
```

### 3. Las redirecciones

Objetivo: trabajar con las redirecciones de canales.

1. El comando **find /** devuelve muchos errores si un simple usuario lo utiliza debido a un problema de permisos. Evite los mensajes de error redireccionándolos hacia un «agujero negro»:

```
$ find / 2>/dev/null
```

2. En el caso anterior y a pesar de los errores, sigue teniendo acceso a muchas ubicaciones y se visualiza una lista muy larga que, por lo tanto, no se puede aprovechar. Coloque esta lista en un fichero llamado `resultado`.

```
$ find / 1>resultado 2>/dev/null
```

3. Ahora, no se visualiza nada. Finalmente, para saber por qué usted no puede acceder a determinados directorios, va a obtener también los mensajes de error en el fichero `resultado`, con la lista de los ficheros. Haga una redirección del canal de error estándar en el canal de salida estándar:

```
$ find / >resultado 2>&1
```

4. No se visualiza nada más. Quiere ambas cosas: un fichero y la visualización de los resultados en pantalla. Se utiliza el comando **tee** con una tubería. Permite recuperar un flujo saliente, colocarlo en un fichero y volver a sacar este flujo como si no hubiera pasado nada:

```
$ find / 2>&1 | tee resultado.
```

Su fichero `resultado` contiene la lista de todos los ficheros accesibles, los errores y el conjunto se visualiza también por pantalla.

### 4. Los filtros y herramientas

Objetivo: trabajar con filtros y herramientas en un fichero clásico.

1. El fichero `/etc/passwd` es un gran clásico bajo Unix. Se compone de siete campos separados por «`:`»: `login:passwd:UID:GID:Comentario:homedir:shell`. Recupere la línea del usuario `root` en `/etc/passwd`:

```
$ grep ^root: /etc/passwd
```

2. En esta línea, recupere el UID de root:

```
$ grep ^root: /etc/passwd | cut -d: -f3
```

3. Cuente el número de usuarios que contiene este fichero usando una redirección en entrada:

```
$ wc -l < /etc/passwd
```

4. Un poco más complicado: recupere la lista de los GID, ordénelos por orden creciente y suprima los duplicados:

```
$ cut -d: -f4 /etc/passwd | sort -n | uniq
```

5. De ahí, extrapolale el número de grupos diferentes utilizados:

```
$ cut -d: -f4 /etc/passwd | sort -n | uniq | wc -l
```

6. Convierta todos los logins a mayúsculas:

```
$ cut -d: -f1 /etc/passwd | tr "[a-z]" "[A-Z]"
```

7. Aísle ahora la octava línea de /etc/passwd. Hay varias soluciones, veamos las dos siguientes:

```
$ head -8 /etc/passwd | tail -1
```

Y:

```
$ grep -n "" /etc/passwd | grep ^8: | cut -d: -f2-
```

## 5. Los procesos

Objetivo: gestionar los procesos.

1. Inicie el proceso sleep 1000 en segundo plano. Recupere su PID.

```
$ sleep 1000&  
[1] 9168
```

(el PID varía, por supuesto).

2. Vuelva a colocar este proceso en primer plano, luego párelo (no lo mate) y de nuevo mándelo a segundo plano.

```
$ fg  
sleep 1000  
(CTRL-Z)  
[1]+ Stopped sleep 1000  
$ bg  
[1]+ sleep 1000 &
```

3. Indique los detalles de este proceso:

```
$ ps p 9168 -f  
UID PID PPID C STIME TTY STAT TIME CMD  
seb 9168 8096 0 10:46 pts/1 S 0:00 sleep 1000
```

4. Modifique la prioridad de este proceso y paselo a un factor 10:

```
$ renice 10 9168  
9168: prioridad anterior 0, nueva prioridad 10
```

5. Liste de nuevo el detalle de este proceso, pero con formato largo. Mire el valor de la columna NI:

```
$ ps p 9168 -l  
F S   UID    PID  PPID  C PRI  NI ADDR SZ WCHAN TTY      TIME CMD  
0 S  1000  9168  8096  0 90  10 -  2324 restar pts/1  0:00  
sleep 1000
```

6. Mande la señal 15 a este proceso. Esto va a terminarlo.

```
$ kill -15 9168  
[1]+  Completado                 sleep 1000
```

## 6. Programación de shell Nivel 1

Objetivo: en un fichero de texto tenemos las líneas siguientes:

1 3

5 7

12 19

...

Escriba un script que acepte este fichero como parámetro, que lo lea y que para cada una de sus líneas calcule la suma de dos números y la muestre con la forma siguiente:

1 + 3 = 4

5 + 7 = 12

12 + 19 = 31

...

1. Verifique al principio del script que el número de parámetros pasado al script es igual a 1 y que este parámetro corresponde efectivamente a un fichero.

```
[ $# -ne 1 -o ! -f $1 ] && exit
```

2. Inicialice una variable a 0, que contendrá el total de cada una de las líneas.

```
result=0
```

3. Se debe leer el fichero línea por línea. Escriba un bucle que lea una línea hasta que se haya alcanzado el final del fichero:

```
while read linea  
do  
  . . .  
done < $1
```

4. En el bucle, recupere los dos valores de las líneas, el separador es el espacio. Coloque dos valores en las variables c1 y c2:

```
while read linea
do
    c1=$(echo $linea | cut -d" " -f1)
    c2=$(echo $linea | cut -d" " -f2)
    ...
done < $1
```

5. Sume estos dos valores y coloque el resultado en result. Visualice result.

```
[ $# -ne 1 -o ! -f $1 ] && exit
result=0
while read linea
do
    c1=$(echo $linea | cut -d" " -f1)
    c2=$(echo $linea | cut -d" " -f2)
    result=$((c1+c2))
    echo -e "$c1 + $c2 = $result"
done < $1
```

## 7. Función Shell

Objetivo: un número es primo cuando sus únicos divisores son 1 y él mismo.

Dicho de otro modo, si se puede dividir un número por otra cosa distinta de uno o este mismo número, y el resultado de esta división es un entero (o el resto de esta división es 0, lo que viene a ser lo mismo), entonces no es primo.

**La cifra 1 no es primo** ya que no tiene dos divisores diferentes.

**La cifra 2 es primo** ( $2 \times 1$ : por lo tanto, dos divisores diferentes).

**Ningún número par** (excepto la cifra 2) es primo (ya que se pueden dividir todos por 2).

Lista de los diez primeros números primos: 2 3 5 7 11 13 17 19 23 29.

Escriba una función llamada **es\_primo**, que coja como parámetro un número y que devuelva el código de retorno 0 si el número es primo, o 1 en el caso contrario.

```
es_primo()
{
    [ $1 -lt 2 ] && return 1
    [ $1 -eq 2 ] && return 0
    [ $(expr $1 % 2) -eq 0 ] && return 1
    i=3
    while [ $((i*i)) -le $1 ]
    do
        [ $(expr $1 % $i) -eq 0 ] && return 1
        i=$((i+2))
    done
    return 0
}
```

# El shell bash

## 1. Papel

Aunque las distribuciones recientes de Linux permiten hacer obviar la introducción de instrucciones de texto al ofrecer entornos gráficos atractivos, un profesional de Linux no puede no conocer el funcionamiento del intérprete de comandos y de los principales comandos asociados.

El intérprete de comandos, o simplemente intérprete, permite ejecutar instrucciones que usted introduce vía teclado o en un script y le devuelve los resultados. Este intérprete es un programa comúnmente llamado shell. Se puede aproximar a la palabra kernel que vimos antes: el kernel significa núcleo. A menudo, está rodeado de una concha dura (piense en un hueso de albaricoque o melocotón). Como «shell» significa concha, viene a decir que es lo que rodea al **núcleo** de Linux: se utiliza mediante comandos. Por lo tanto, es una interfaz que funciona en modo texto entre el núcleo de Linux y los usuarios (avanzados) o las aplicaciones.

Hay varios shells: cada uno dispone de especificaciones propias. El Bourne Shell (sh) es el shell más conocido y más habitual en los Unix. El C-Shell (csh) retoma la estructura del lenguaje C. El Korn Shell (ksh) es una evolución del Bourne Shell. El Z-Shell (zsh) es a su vez una evolución del Korn Shell. El shell de referencia en Linux se llama Bourne Again Shell (bash). A continuación le presentamos una lista exhaustiva de intérpretes de comandos que puede encontrar en Linux:

- sh: Thompson Shell (ya no existe);
- sh: Bourne Shell (sustituyó al anterior);
- bash: Bourne Again Shell;
- ksh: Korn Shell;
- csh: C Shell;
- zsh: Z Shell;
- tcsh: Tenex C Shell;
- ash: A Shell;
- dash: Debian Almquist Shell.

---

 La lista de los shells actualmente presentes en su instalación de Linux se encuentra en el fichero /etc/shells.

---

## 2. Bash: el shell por defecto

### a. Un shell potente y libre

El bash es un derivado de Bourne Shell. Bourne es el nombre del principal programador de este shell. La expresión «Bourne Again» es un guiño a los orígenes del bash (Bourne) y un juego de palabras en «I born again», lo que significa «he nacido otra vez» o «reencarnado». El bash retoma sh pero también funcionalidades de ksh o csh.

El bash no está presente únicamente bajo Linux. Al ser un programa libre, se puede compilar o ejecutar en numerosas plataformas. Es el shell de referencia en los sistemas MacOS y existe también para Windows.

El shell funciona en un terminal. Originalmente, un terminal es una verdadera máquina que sólo dispone de

lo necesario para entrar instrucciones (el teclado) y visualizar los resultados (una pantalla o incluso hace mucho tiempo una simple impresora con papel listado). Aunque en la actualidad aún existan terminales físicos, en máquinas corrientes han sido sustituidos por programas que emulan terminales. Se distinguen dos tipos de ellos en Linux:

- las consolas virtuales de texto, el modo por defecto de Linux cuando arranca o funciona bajo entorno gráfico;
- las consolas o terminales gráficos, como xterm, eterm o konsole, que son emuladores de terminales en el seno de ventanas gráficas.

El shell funciona en un terminal. Espera entradas por teclado en la consola o la ventana, y visualiza sus resultados en el mismo lugar. Cualquier usuario avanzado de Linux o Unix en general tiene al menos un terminal abierto casi permanentemente. La apertura de un terminal (o consola, en este caso estas palabras son sinónimas) ejecuta automáticamente el shell por defecto.

## b. Línea de comandos

El shell espera entradas por el teclado en una línea llamada línea de comandos o prompt. Un cursor, representado por un rectángulo fijo, intermitente o un carácter subrayado, indica la posición actual de su entrada.

La línea (prompt) proporciona información en el terminal y su posición en el sistema de ficheros.

```
seb@slyserver:/home/public>
```

En esta clásica línea, obtiene cuatro datos:

- seb: es el nombre de conexión o login del usuario actualmente conectado al terminal;
- slyserver: es el nombre de anfitrión (hostname), el nombre lógico de la máquina conectada al terminal;
- /home/public: es la posición actual del shell en el sistema de ficheros;
- >: es la terminación estándar del bash para un usuario sin privilegios.

Esta línea le informa de que es el usuario sin privilegios de administración seb quien utiliza el terminal (está conectado) en la máquina slyserver y que se encuentra actualmente en /home/public.

El carácter de terminación puede tener otros significados:

- \$ indica que el usuario no tiene privilegios particulares, como con >.
- # indica que el usuario es el administrador root que tiene todos los privilegios.

La ruta puede variar:

seb@slyserver:~>: el carácter tilde ~ indica que se encuentra en su directorio personal.

seb@slyserver:~/test>: al combinarse con ~ (su directorio personal), indica que está en el subdirectorio test de éste.

Para cualquier otra ruta se suele sustituir la línea de comandos por un simple símbolo de dólar, \$, con el fin de ganar espacio.

## 3. Utilizar el shell

## a. La introducción de datos

En el terminal, el teclado se utiliza de forma intuitiva. Puede desplazarse en la línea con las flechas de derecha e izquierda del teclado y borrar caracteres con las teclas [Tab] y [Supr]. Ejecute el comando que ha introducido presionando la tecla [Intro].

Ha llegado la hora de probar comandos. El comando **date** indica la fecha y la hora actuales. Por supuesto no obtendrá el mismo resultado, y no siempre en el mismo idioma; depende de su instalación de Linux.

```
$ date  
vier feb 22 21:28:12 CET 2008
```

Un comando práctico, **pwd**, permite saber en qué directorio se encuentra usted.

```
$ pwd  
/home/seb
```

shell indica que está en el directorio /home/seb.

## b. Sintaxis general de los comandos

Los comandos o instrucciones (las dos palabras son sinónimas en este caso) GNU tienen a menudo una sintaxis con la misma estructura:

Comando [parámetros] [argumentos]

Un comando puede no tener ni parámetros ni argumentos. En este caso, ejecuta la acción por defecto para la cual se programó, o muestra un mensaje de error, de haberlo.

Un parámetro es una opción del comando. Las dos palabras son sinónimas aquí. A menudo es una simple letra o una simple cifra precedida de un guión: `-l`, `-p`, `-s`, etc. Si el comando acepta varios parámetros, usted los puede introducir unos tras otros separándolos por espacio: `-l -r -t`, o escribiendo un solo guión y luego todos los parámetros: `-lrt`. Se aceptan las dos sintaxis: producen el mismo resultado. Simplemente, la segunda es más corta.

 En ciertos casos, un parámetro necesita un argumento, por ejemplo un nombre de fichero. En este caso, es preferible separar este parámetro de los demás: `-lrt -f mifichero`.

Los argumentos son los valores en los que un comando debe ejecutar su acción. El tipo de valor (fichero, texto, números) depende del comando.

## c. Primer ejemplo concreto con cal

Tomemos el ejemplo del comando **cal**. Éste admite varios parámetros y argumentos. Si se le invoca sin argumentos, muestra el calendario del mes en curso y resalta el día actual.

```
$ cal  
febrero 2008  
do lu ma mié ju vie sá  
1 2  
3 4 5 6 7 8 9  
10 11 12 13 14 15 16  
17 18 19 20 21 22 23  
24 25 26 27 28 29
```

comando **cal** admite dos argumentosopcionales. Si se precisa sólo uno, se trata del año, y se muestra el calendario de este año en su totalidad. Si se le pasan dos argumentos, el primero es el mes; el segundo, el año.

```
$ cal 12 1975
diciembre 1975
do lu ma me ju vie sa
  1  2   3   4   5   6
  7  8   9   10  11  12  13
14 15  16  17  18  19  20
21 22  23  24  25  26  27
28 29  30  31
```

comando admite también unos parámetros. Observará que por defecto se prevé la visualización para los anglosajones: la primera columna es un domingo, que representa el primer día de la semana. En España, es el lunes. El parámetro **-m** (de monday, en inglés) permite precisarlo:

```
$ cal -m 12 1975
diciembre 1975
lu ma me ju vie sa do
  1  2   3   4   5   6   7
  8  9   10  11  12  13  14
15 16  17  18  19  20  21
22 23  24  25  26  27  28
29 30  31
```

Un segundo parámetro **-3** permite visualizar los meses que preceden y siguen al mes determinado (o el mes en curso).

```
seb@slyserver:~> cal -m -3 12 1975
noviembre 1975      diciembre 1975      enero 1976
lu ma me ju vie sa do  lu ma me ju vie sa do  lu ma me ju vie sa do
  1  2   3   4   5   6   7   1  2   3   4   5   6   7   1  2   3   4
  3  4   5   6   7   8   9   8  9   10  11  12  13  14   5  6   7   8   9  10  11
10 11  12  13  14  15  16  15 16  17  18  19  20  21  12 13  14  15  16  17  18
17 18  19  20  21  22  23  22 23  24  25  26  27  28  19 20  21  22  23  24  25
24 25  26  27  28  29  30  29 30  31  26 27  28  29  30  31
```

como puede agrupar los parámetros, el comando siguiente produce el mismo resultado.

```
$ cal -m3 12 1975
```

## d. Encadenar los comandos

Puede ejecutar varios comandos en una sola línea, los unos tras los otros. Para ello, basta con separarlos con un punto y coma.

```
$ date;pwd;cal
lun feb 25 22:29:09 CET 2008
/usr/share/man/man9
  febrero 2008
do lu ma me ju vie sa
  1  2
  3  4  5   6   7   8   9
10 11 12  13  14  15  16
17 18 19  20  21  22  23
24 25 26  27  28  29
```

## e. Visualizar texto con echo

No hay nada más sencillo que visualizar texto. El comando **echo** está hecho para ello. Como casi todos los comandos, acepta los parámetros, además de los argumentos, en forma de texto. Para visualizar un texto sencillo:

```
$ echo Hola amigos  
Hola amigos
```

Puede colocar el texto entre comillas o simples apóstrofes para clarificar y agrupar el texto a visualizar. Podrá ver más adelante que esto tiene un significado particular.

Observe que, por defecto, su texto se visualiza y echo efectúa solo un retorno de carro para empezar otra línea. Puede modificar su texto para añadirle secuencias de caracteres que tienen una acción particular. Si ya conoce el lenguaje C, son las mismas. En la lista se encuentran sólo las más utilizadas:

Secuencia	Acción
\n	Pasar a la línea
\t	Tabulación horizontal
\c	Suprimir el salto de línea final
\b	Retorno de un carácter atrás
\\"	Visualiza la contrabarra (barra oblicua inversa)
\nnn	Visualiza el carácter especificado en octal

Para utilizar estas secuencias, añada el argumento **-e**:

```
$ echo -e "Hola.\tMe llamo Seb\b\b\bNadie\n"  
Hola. Me llamo Nadie
```

## f. Comandos internos y externos

Existen dos tipos de comandos:

- Los comandos externos son programas binarios presentes como ficheros en su disco duro (o cualquier otro soporte de datos). Cuando ejecuta el comando, se carga este fichero en memoria y se inicia como proceso (esta noción se explicará en este mismo capítulo).
- Los comandos internos son internos al shell y se ejecutan en el interior de él. Estos comandos forman parte del programa shell, el bash. Los comandos **cd** o **pwd** son dos ejemplos de comandos internos. Cuando los ejecuta, el shell ejecuta las funciones definidas en su interior que les corresponden.

Puede distinguir un comando interno de un comando externo con la ayuda del comando interno **type**. Así, **date** es un comando externo. Puede observar que es un fichero presente en **/bin**, mientras que **pwd** es interno al shell.

```
$ type date  
date is hashed (/bin/date)
```

```
$ type pwd  
pwd is a shell builtin
```

► También puede encontrar otros tipos, como los alias de comandos que son atajos de comandos propios del shell. Así el shell bash de ciertas distribuciones Linux proponen alias como ll, que corresponde en realidad a ls -l.

```
$ type ll  
ll is aliased to `ls -l'
```

### g. Algunos atajos útiles

Se deben conocer algunas secuencias de atajos de comandos:

- **[Ctrl] C**: interrupción del programa: se termina.
- **[Ctrl] Z**: para el programa (ver los procesos).
- **[Ctrl] D**: interrumpe una introducción de datos en un prompt >.

## 4. El historial de comandos

Le resultará muy útil poder volver a llamar a un comando que ya ejecutó navegando por el historial de los comandos con las teclas [Flecha arriba] y [Flecha abajo]. La flecha arriba vuelve atrás en el historial. Si ha tecleado los dos comandos anteriores (**date** y luego **pwd**), al presionar una primera vez en la flecha arriba se visualiza en la línea de comandos **pwd**, y a la segunda se visualiza el comando **date**. La flecha abajo navega en el otro sentido hasta la línea de origen. Si pulsa la tecla [Intro] vuelve a ejecutar el comando.

Cuantos más comandos teclee, más se amplia el historial. El shell conserva así un gran número de entradas en el historial (se puede modificar el número de líneas guardadas). Se conserva este historial en un fichero caché de su directorio personal llamado `.bash_history`. Puede ver el contenido del historial con el comando **history**. El resultado siguiente está truncado de manera voluntaria, ya que la lista es demasiado larga.

```
$ history  
...  
1000 date  
1001 pwd  
1002 uname -a  
1003 ls  
1004 fc -l -5  
1005 history
```

comando **fc** funciona como **history** cuando se utiliza con el parámetro `-l`. Por defecto, se limita a los últimos quince comandos. También puede pasar el número de últimos comandos, como a continuación para los diez últimos:

```
$ fc -l -10  
995 ssh -X seb@192.168.1.130  
996 fc -l  
997 fc -l -20  
998 ls  
999 pwd  
1000 cd  
1001 uname -a  
1002 fc -l  
1003 cat /etc/passwd
```

```
1004      ls -lrtR
```

Puede volver a llamar a un comando con fc y el parámetro `-s` seguido del número del comando. Entonces se ejecutará de manera automática.

```
$ fc -s 1001
uname -a
Linux slyserver 2.6.22.17-0.1-default #1 SMP 2008/02/10 20:01:04 UTC
x86_64 x86_64 x86_64 GNU/Linux
```

Finalmente, puede sustituir un elemento del comando por otro antes de ejecutarlo. Por ejemplo, puede sustituir `fc` por `ls` en la entrada 1002 del historial:

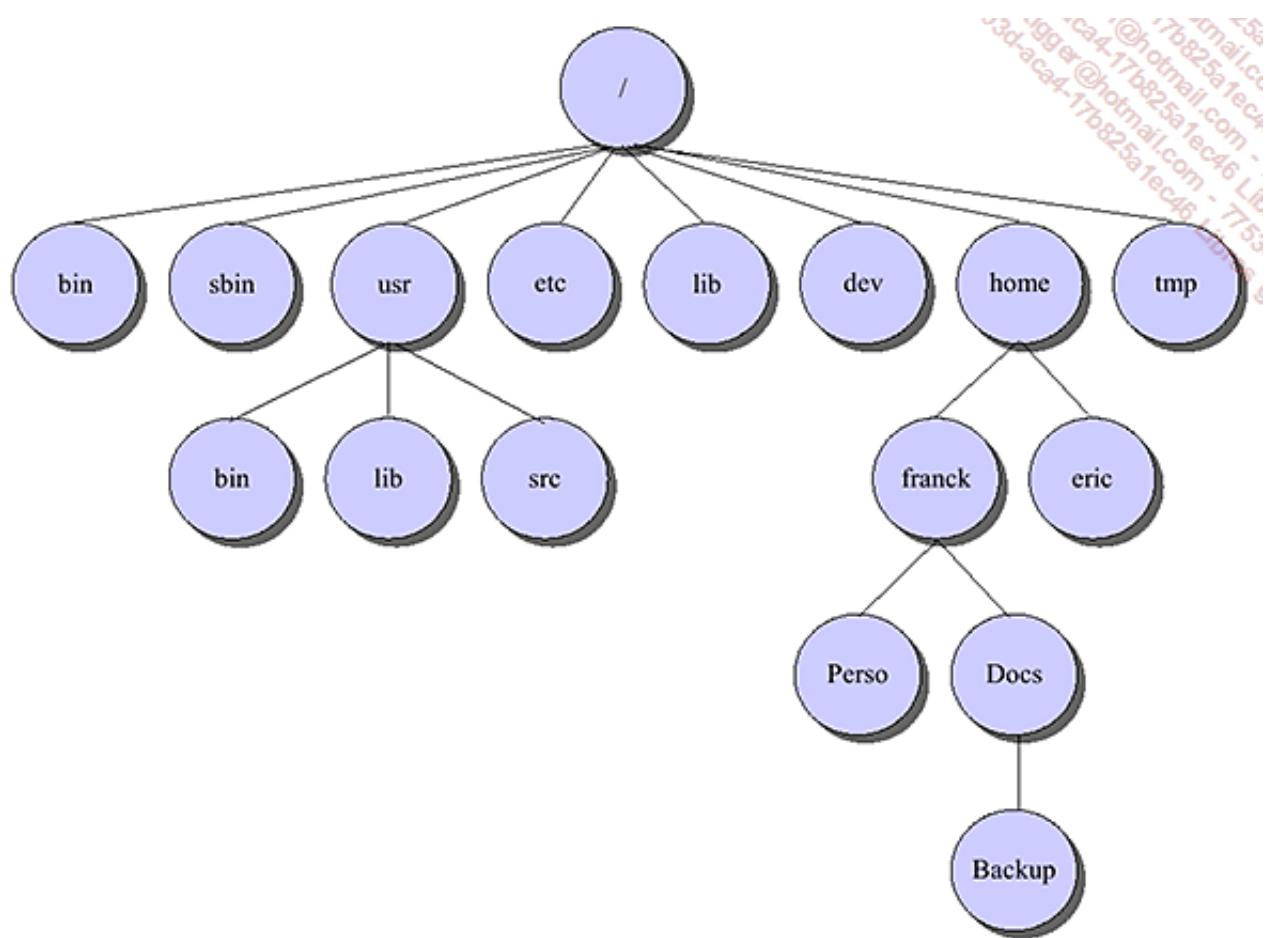
```
$ fc -s fc=ls 1002
ls -l
total 775944
-rw-r--r-- 1 seb users      15391 may 14  2007 AR-1179161176460.pdf
...
```

## **La gestión de los ficheros**

## 1. El sistema de ficheros

Un sistema de ficheros, llamado comúnmente File System o FS, determina la organización de los datos en un soporte de almacenamiento, y por lo tanto cómo el sistema operativo gestiona y organiza los ficheros.

Linux es, como todo Unix, un sistema operativo completamente orientado a ficheros. Se representa todo (o casi todo) con un fichero, tanto los datos (ficheros de datos de cualquier tipo, como una imagen o un programa) como los periféricos (terminales, ratones, teclado, tarjeta sonido, etc.) o aun los medios de comunicación (sockets, tuberías nombradas, etc.). Se puede decir que el sistema de ficheros es el corazón de cualquier sistema Unix.



## *Ejemplo de estructura en árbol de Linux*

El sistema de ficheros de Linux es jerárquico. Describe un árbol de directorios y subdirectorios, a partir de un elemento básico llamado raíz o root directory.

## 2. Los diferentes tipos de ficheros

Distinguimos tres tipos de ficheros: ordinarios, catálogo, especiales.

#### **a. Los ficheros ordinarios o regulares**

Los ficheros ordinarios se llaman también ficheros regulares, ordinary files o regular files. Son ficheros totalmente clásicos que contienen datos. Por datos se debe entender cualquier contenido:

- texto;
- imagen;
- audio;
- programa binario compilado;
- script;
- base de datos;
- librería de programación;
- etc.

Por defecto, nada permite diferenciar unos de otros, salvo la utilización de algunas opciones de determinados comandos (`ls -F` por ejemplo) o el comando **file**.

```
$ file nom_fic
nom_fic: 32 Bits ELF Executable Binary (stripped)
```

 Linux desconoce la noción de extensión de fichero como componente interno de la estructura del sistema de ficheros. Dicho de otro modo, una extensión no es relevante dentro de un sistema de ficheros y se la considera simplemente como parte del nombre. Sólo sirve para distinguir visual y rápidamente el posible contenido de un fichero en comparación con otro.

Como Linux no gestiona las extensiones, el nombre de un programa no termina casi nunca por un «.exe»; habrá que encontrar otro método para distinguirlo.

## b. Los catálogos

Los ficheros catálogo son los directorios o carpetas. Los directorios permiten organizar el disco duro creando una jerarquía. Un directorio puede contener ficheros normales, ficheros especiales y otros directorios de manera recursiva.

Un directorio no es más que un fichero particular que contiene la lista de los propios ficheros presentes en este directorio. Esta noción resultará muy útil cuando se trate el tema de los permisos.

## c. Los ficheros especiales

El tercer tipo de ficheros es el especial. Existen varios tipos de ficheros especiales. Por ejemplo, los drivers de los periféricos están representados por ficheros especiales de la carpeta `/dev`.

Son principalmente ficheros que sirven de interfaz para los diversos periféricos. Se pueden utilizar, según el caso, como ficheros normales. Cuando se accede en modo lectura o escritura a estos ficheros se redirigen hacia el periférico (pasando por el driver asociado si existe). Por ejemplo, si dirige un fichero de onda sonora (wave) hacia el fichero que representa la salida de la tarjeta sonido, hay muchas probabilidades que este sonido sea audible por sus altavoces.

# 3. Nomenclatura de los ficheros

No se puede dar cualquier nombre a un fichero; hay que seguir unas simples reglas, válidas para todos los

tipos de ficheros.

En cuanto a los antiguos sistemas Unix, un nombre de fichero no podía superar los 14 caracteres. En los sistemas actuales, Linux incluido, se puede llegar hasta 255 caracteres. La posible extensión está incluida en la longitud del nombre del fichero.

Un punto extremadamente importante: Linux respeta la distinción entre los nombres de ficheros en minúsculas y en mayúsculas. Pepito, PEPITO, PePito y pepito son nombres de ficheros diferentes, con un contenido diferente.

Se acepta la mayoría de los caracteres (las cifras, las letras, las mayúsculas, las minúsculas, ciertos signos, los caracteres acentuados), incluyendo el espacio. Sin embargo, se deben evitar algunos caracteres, ya que tienen un significado particular dentro del shell: & ; ( ) ~ <espacio> \ / | ` ? - (al principio del nombre).

Los nombres siguientes son válidos:

- Fichero1
- Paga.txt
- 123tratamiento.sh
- Paga\_junio\_2002.xls
- 8

Los nombres siguientes pueden crear problemas:

- Fichero\*
- Pago(diciembre)
- Ben&Nuts
- Paga junio 2002.xls
- -f

## 4. Las rutas

### a. Estructura y nombre de ruta

Las rutas permiten definir una ubicación en el sistema de ficheros. Es la lista de los directorios y subdirectorios utilizados para acceder a un sitio determinado de la estructura hasta la posición deseada (directorio, fichero). Se suele completar un nombre de fichero con su ruta de acceso. Por eso el fichero pepito del directorio dir1 es diferente del fichero pepito del directorio dir2. Al ser jerárquico el sistema de ficheros de Unix, tiene forma de estructura en árbol.

El esquema presentado en la sección La gestión de los ficheros - El sistema de ficheros de este capítulo representa una estructura en árbol de un sistema de ficheros Linux. La / situada arriba del todo se llama raíz o root directory (no confundir con el directorio del administrador root). El nombre de la ruta o path name de un fichero es la concatenación, desde la raíz, de todos los directorios que se deben cruzar para acceder a él, que están separados cada uno por el carácter /. Es una ruta absoluta, como la siguiente:

/home/pepito/Docs/Backup/fic.bak

Una ruta absoluta o completa:

- empieza desde la raíz. Por lo tanto, comienza con una /,

- describe todos los directorios que hay que cruzar para acceder al sitio deseado,
- no contiene ni . ni ..

## b. Directorio personal

Al crear un nuevo usuario, el administrador le asigna un directorio personal llamado home directory. Cuando se conecta, el usuario es dirigido directamente a ese directorio, que es el suyo personal. En este directorio el usuario podrá crear sus propios ficheros y subdirectorios.

```
Login: seb
Password: xxxxxxxxxxxx
$ pwd
/home/seb
```

## c. Ruta relativa

Un nombre de ruta también puede ser relativo a su posición en el directorio actual. El sistema (o el shell) recuerda la posición actual de un usuario en el sistema de ficheros, el directorio activo. Usted puede acceder a otro directorio de la estructura desde su ubicación actual sin teclear la ruta completa, con sólo precisar la ruta más corta en relación con su posición actual dentro de la estructura.

Para ello, hace falta a menudo utilizar dos entradas particulares de directorios:

- El punto . representa el directorio corriente, activo. Suele estar implícito.
- Los dos puntos .. representan el directorio de nivel superior.

Una ruta relativa:

- describe una ruta relativa a una posición determinada en la estructura, en general (pero no siempre) desde la posición actual;
- describe en principio la ruta más corta para ir de un punto a otro;
- puede contener puntos o dos puntos.

Las tres afirmaciones anteriores no son obligatorias:

- /usr/local/bin es una ruta completa o absoluta;
- Documents/Photos es una ruta relativa: se considera que existe el directorio Documents en el directorio corriente;
- ./Documents/Photos es una ruta relativa perfectamente idéntica a la anterior, con la salvedad de que el punto indica el directorio activo (corriente) de manera explícita. «./Documents» indica de manera explícita el directorio Documents en el directorio activo;
- /usr/local/..../bin es una ruta relativa: los .. son relativos a /usr/local y suben un nivel hacia /usr. La ruta final es, por lo tanto /usr/bin.

## d. La virgulilla

El bash interpreta el carácter virgulilla ~ como un alias del directorio personal. Las rutas pueden ser relativas a la virgulilla, pero ésta no debe ir precedida por carácter alguno. Para desplazarse en el directorio tmp de su carpeta personal esté donde esté: \$ cd ~/tmp

Si introduce esto, obtendrá un error: \$ cd ~/

## e. cd

Para desplazarse por los directorios, utiliza el comando **cd** (*change directory*). El comando **pwd** (*print working directory*), que usted ya hemos comentado, muestra la ruta completa del directorio actual.

Si introduce **cd ..**, no se mueve. El punto será muy útil cuando tenga que especificar rutas explícitas a comandos ubicados en el directorio donde usted está ubicado.

**Cd ..** sube un nivel. Si se encontraba en /home/seb, ahora estará en home.

El comando **cd** sin argumento permite volver directamente a su directorio de usuario.

A continuación, presentamos un ejemplo. El usuario seb se encuentra en su directorio personal. Se mueve mediante una ruta relativa hacia /home/public. Con .. sube hacia /home, por lo tanto con ../public se mueve a /home/public. De ahí, vía una ruta completa, se dirige hacia /usr/local/bin, y luego decide, con la ayuda de una ruta relativa, ir a /usr/lib: el primer .. baja hacia usr/local, el segundo hacia /usr, y luego vuelve hacia /usr/lib. Finalmente, seb vuelve a su directorio personal con **cd** sin argumento. Aquí se da la línea completa para una mejor comprensión.

```
seb@slyserver:~> pwd  
/home/seb  
seb@slyserver:~> cd ../public  
seb@slyserver:/home/public> cd /usr/local/bin  
seb@slyserver:/usr/local/bin> cd ../../lib  
seb@slyserver:/usr/lib> cd  
seb@slyserver:~>
```

## 5. Los comandos básicos

### a. Listar los ficheros y los directorios

El comando **ls** permite listar el contenido de un directorio (catálogo) en líneas o columnas. Soporta varios parámetros, de los cuales los más importantes son:

Parámetro	Significado
-l	Para cada fichero o carpeta, facilita información detallada.
-a	Se visualizan los ficheros escondidos (empiezan por un punto).
-d	En un directorio, precisa el propio directorio, y no su contenido.
-F	Añade un carácter al final del nombre para especificar el tipo: / para un directorio, * para un ejecutable, @ para un vínculo simbólico, etc.
-R	Si el comando detecta subdirectorios, entra en ellos de manera recursiva.
-t	Se filtra la salida por fecha de modificación del más reciente al más antiguo. Se visualiza esta fecha.
-c	Muestra/ordena (con -t) por fecha de cambio de estado del fichero.
-u	Muestra/ordena (con -t) por fecha de acceso del fichero.

-r	Se invierte el orden de salida.
-i	Muestra el inodo del fichero.
-C	La visualización se hace sobre varias columnas (por defecto).
-1	La visualización se hace sobre una sola columna.

El parámetro que le facilita más información es -1: proporciona ciertos detalles relativos a los ficheros.

```
$ ls -l
total 4568
-rw-r--r-- 1 seb users 69120 sep 3 2006 3i_recuperación_2006.doc
-rw-r--r-- 1 seb users 9632 sep 3 2006 3i_recuperación_2006.odt
-rw-r--r-- 1 seb users 6849 nov 17 2003 control_1I2_mártes.sxw
...
```

La línea de salida indica el tamaño total en kilobytes del contenido del directorio. Este tamaño representa el conjunto de los ficheros ordinarios del directorio y no tiene en cuenta los posibles subdirectorios y su contenido (para ello, habrá que utilizar el comando du).

Luego viene la lista detallada de todo el contenido.

-rw-r--r--	1	seb	users	69120	sep 3 2006	3i_recuperación_2006.doc
1	2	3	4	5	6	7

- 1: El primer símbolo representa el tipo de fichero (-: ordinario, d: directorio, l: vínculo simbólico...); los otros, por bloques de tres, los derechos para el usuario (rw-), el grupo (r--) y todos (r--). Se explican los derechos en el capítulo Los discos y el sistema de ficheros.
- 2: Un contador de vínculos (capítulo Los discos y el sistema de ficheros).
- 3: El propietario del fichero, que suele ser su creador.
- 4: El grupo al cual pertenece el fichero.
- 5: El tamaño del fichero en bytes.
- 6: La fecha de la última modificación (a veces con la hora), siguiendo el parámetro (t, c, u).
- 7: El nombre del fichero.

Le puede resultar muy útil la posibilidad de listar sus ficheros de tal manera que se visualicen al final de la lista los modificados más recientemente. Así, en caso de presencia de un gran número de ficheros, tendrá delante estos últimos. La ordenación por fecha de modificación se hace con -t, y en el orden contrario, con -r. Añádale los detalles con -1.

```
$ ls -lrt
-rw-r--r-- 1 seb users 66107 ene 9 17:24 Parcial_1_1I_2008.pdf
-rw-r--r-- 1 seb users 13777 ene 10 17:58 parcial_3I_ppa_2007.odt
-rw-r--r-- 1 seb users 64095 ene 10 17:58 parcial_3I_ppa_2007.pdf
-rw-r--r-- 1 seb users 100092 feb 22 22:21 curso_shell_unix.odt
```

 ls -1 -r -t es estrictamente idéntico a ls -lrt, como ya se ha indicado en la sintaxis general de los comandos.

---

Un recurso mnemotécnico para recordar esta secuencia de argumentos es utilizarla bajo la forma `-rtl`(el orden de los argumentos no tiene importancia aquí) y pensar en la famosa radio europea RTL.

## b. Gestionar los ficheros y los directorios

### Crear ficheros vacíos

Puede necesitar crear ficheros vacíos para hacer pruebas. Un comando práctico para ello es **touch**. Utilizado con el nombre de un fichero como argumento únicamente, crea un fichero con un tamaño cero.

```
$ touch fictest  
$ ls -l fictest  
-rw-r--r-- 1 seb users 0 feb 29 15:13 fictest
```

La creación de ficheros vacíos no es el principal uso de touch originariamente. Si vuelve a ejecutar el mismo comando en el fichero, observará que la fecha de modificación ha cambiado. El manual de touch le informará de que así es posible modificar completamente la fecha y la hora de un fichero. Esto puede ser útil para forzar las copias de seguridad incrementales de ficheros.

### Crear directorios

El comando **mkdir** (*make directory*) permite crear uno o varios directorios, o una estructura completa. Por defecto, el comando no crea una estructura. Si pasa como argumentos `dir1/dir2` y `dir1` no existe, el comando devuelve un error. En este caso, utilice el parámetro `-p`.

```
mkdir [-p] dir1 [dir2] ... [dirn]
```

```
$ mkdir Documentos  
$ mkdir Documentos/Fotos  
$ mkdir -p Archivos/antiguallas  
$ ls -R  
. :  
Archivos Documentos fictest  
  
. /Archivos:  
antiguallas  
  
. /Archivos/antiguallas:  
  
. /Documentos:  
Fotos
```

### Suprimir directorios

El comando **rmdir** (*remove directory*) suprime uno o varios directorios. No puede suprimir una estructura. Si el directorio que pretende eliminar contiene ficheros o directorios, el comando devuelve un error. Por lo tanto, el directorio no debe contener ni ficheros ni directorios, y ello aunque los propios subdirectorios estén vacíos.

```
rmdir dir1 [dir2] ... [dirn]
```

---

 No hay parámetro `-r` (recursividad) para el comando **rmdir**. Para suprimir una estructura tendrá que utilizar el comando **rm**.

---

```
$ rmdir Documentos/  
rmdir: Documentos/: El directorio no está vacío.  
$ rmdir Documentos/Fotos  
$
```

## **Copiar ficheros**

El comando **cp** (copy) copia uno o varios ficheros en otro fichero o en un directorio.

```
cp fic1 [fic2 ... ficn] Destino
```

En el primer caso, se vuelve a copiar fic1 en Destino. Si Destino existe, se sobreescribe sin aviso según el parámetro pasado y según los permisos. En el segundo caso, se copian de nuevo fic1, fic2 y así sucesivamente en el directorio Destino. Las rutas pueden ser absolutas o relativas. El comando puede tomar, entre otras, las opciones siguientes:

Parámetro	Significado
-i	Pide confirmación de copia para cada fichero.
-r	Recursivo: copia un directorio y todo su contenido.
-p	Se preservan los permisos y fechas.
-f	Forzar la copia.

Le debe de llamar la atención el funcionamiento de cp con las copias de directorios. El funcionamiento es diferente según exista o no el directorio de destino. En el primer caso, dir2 no existe. Se copia el directorio dir1 en dir2. Al final dir2 es una copia exacta de dir1.

```
$ ls -d dir2  
ls: no puede acceder a dir2: No existe el fichero o directorio  
$ cp -r dir1 dir2  
$ ls  
dir1 dir2
```

Ahora que dir2 existe, ejecute de nuevo el comando cp. Esta vez, como dir2 existe, no será sobreescrito, como cabía esperar. El comando determina que, al ser dir2 el destino, se debe copiar dir1 en el destino: se copia dir1 en dir2.

```
$ cp -r dir1 dir2  
$ ls dir2  
dir1
```

## **Mover y volver a nombrar un fichero**

El comando **mv** (move) permite mover, volver a nombrar un fichero o las dos cosas a la vez. Funciona como el comando cp. Los parámetros -f e -i tienen el mismo efecto. Con los tres comandos mv sucesivos siguientes:

- se vuelve a nombrar txt1 como txt1.old;
- se mueve txt2 a dir1;

- se mueve txt3 a dir1 y se vuelve a nombrar como txt3.old.

```
$ touch txt1 txt2 txt3
$ mv txt1 txt1.old
$ mv txt2 dir1/txt2
$ mv txt3 dir1/txt3.old
```

Observe la existencia del parámetro **-u**: si el fichero de destino es más reciente, impide que se sobreescriba.

### **Suprimir un fichero o una estructura**

El comando **rm** (*remove*) suprime uno o varios ficheros y, si es preciso, una estructura completa, según las opciones. La supresión es definitiva.

```
rm [Opciones] fic1 [fic2...]
```

Las opciones son las habituales, pero dada la particularidad y la peligrosidad del comando, parece necesario repasarlas.

Parámetro	Significado
<b>-i</b>	El comando requerirá una confirmación para cada uno de los ficheros a suprimir. Según la versión de Unix, el mensaje cambia y la respuesta también: y, Y, O, o, N, n, a veces todas.
<b>-r</b>	El parámetro siguiente que se espera es un directorio. En este caso, la supresión es recursiva: se suprimen todos los niveles inferiores, tanto los directorios como los ficheros.
<b>-f</b>	Fuerza la supresión.

Por orden, los comandos siguientes suprimen un simple fichero, suprimen un directorio y suprimen una estructura de manera forzada:

```
$ rm fic1
$ rm -r dir1
$ rm -rf /home/public/depots
```

 El uso combinado de los parámetros **-r** y **-f**, aunque muy útil y práctico, es muy peligroso, en particular como root. No se le pide ninguna confirmación. Al menos que se utilicen herramientas de recuperación de datos específicos, caras y poco eficaces, se perderán sus datos de manera irremediable.

Permítanos ponerle en un atolladero. Suponga que dispone de un fichero cuyo nombre comienza con un guion. ¿Es posible suprimirlo con **rm**?

```
$ >-i # ver las redirecciones
$ rm -i
rm: missing operand
Pruebe: `rm --help' para más información.
```

Es imposible suprimir el fichero «-i» de esta manera, ya que **rm** lo interpreta como un parámetro, y no como un argumento. Por lo tanto hay que actuar con astucia. Existen dos soluciones:

- Utilizar la opción **GNU --** que significa el final de los parámetros y el principio de los argumentos.

- Añadir una ruta, relativa o completa, antes del guión.

Esta última solución tiene la ventaja de ser un estándar. Las dos líneas son equivalentes:

```
$ rm -- -i
$ rm ./-i
```

### **Los vínculos simbólicos**

Puede crear vínculos simbólicos, que son un poco como atajos a un fichero. Un vínculo es un fichero especial que contiene como información la ruta hacia otro fichero. Es un tipo de alias. Existen dos tipos de vínculos: el vínculo duro (hard link), que veremos más adelante, durante el estudio de los sistemas de ficheros, y el vínculo simbólico (soft link), que corresponde a la definición dada.

Es posible crear vínculos simbólicos hacia cualquier tipo de fichero, sea cual sea y esté donde esté. El comando de creación de vínculos simbólicos no comprueba la existencia del fichero al que se apunta. Es posible crear vínculos a ficheros que no existen.

```
ln -s fichero vínculo
```

Si fuera necesario, el vínculo se comportará como el fichero al que se apunta, con los mismos permisos y las mismas propiedades:

- si el fichero al que se apunta es un programa, ejecutar el vínculo lleva a ejecutar el programa;
- si el fichero al que se apunta es un directorio, un cd sobre el vínculo entra en este directorio;
- si el fichero al que se apunta es un fichero especial (periférico), se ve el vínculo como periférico;
- etc.

Sólo en caso de que se elimine, el vínculo simbólico se «separa» del fichero al que apunta. La supresión de un vínculo simbólico implica la supresión de este vínculo únicamente, y no del fichero al que apunta. La supresión del fichero al que se apunta no implica la supresión de los vínculos simbólicos asociados. En este caso, el vínculo apunta al vacío.

```
$ touch fic1
$ ln -s fic1 vínculofic1
$ ls -l
-rw-r--r-- 1 seb users    0 mar  4 19:16 fic1
lrwxrwxrwx 1 seb users    4 mar  4 19:17 vínculofic1 -> fic1
$ ls -F
fic1 vínculofic1@
$ echo titi>fic1
$ cat vínculofic1
titi
```

Este ejemplo muestra que un vínculo simbólico es en realidad un fichero especial de tipo «l» que apunta hacia otro fichero. Observe en la lista detallada la presencia de una flecha que indica sobre qué fichero apunta el vínculo. Se distingue el carácter @ al indicar que se trata de un vínculo simbólico durante la utilización del parámetro -F. Si usted dispone de un terminal de color, es posible que el vínculo simbólico aparezca en azul claro (por convención en Linux). Si aparece en rojo, es que apunta al vacío.

---

 Que un vínculo apunte al vacío no significa que no apunte a nada. Quizá esté hecho a propósito: es posible crear vínculos hacia puertos USB, o CD-ROM, entre otros, detrás de los cuales hay sistemas de ficheros removibles. En este caso, el vínculo se vuelve activo cuando se inserta el soporte o se monta el sistema de ficheros removido.

---

Se explicará más adelante el comando **echo** y el signo **>**. El efecto aquí es la escritura en el fichero **fic1** de «**titi**». El comando **cat** visualiza el contenido de un fichero. Al representar el vínculo **fic1**, la salida es la esperada.

 Cuidado, los permisos indicados son los del fichero especial y no tienen otro significado: no significa que todo el mundo tiene derechos en el fichero al que se apunta. Durante su utilización, son los derechos del fichero o de la carpeta a los que se apuntan los que prevalecen.

### c. Comodines: carácter de sustitución

Durante la utilización de comandos en relación con el sistema de ficheros, puede resultar interesante filtrar la salida de nombres de ficheros con la ayuda de determinados criterios, por ejemplo con el comando **ls**. En vez de visualizar toda la lista de los ficheros, se puede filtrar la visualización de varios criterios y caracteres especiales.

Caracteres(s)	Papel
*	Sustituye una cadena de longitud variable, incluso vacía.
?	Sustituye cualquier carácter único.
[...]	Una serie o un rango de caracteres.
[a–b]	Un carácter entre el rango indicado (de a a b incluida).
[!...]	Inversión de la búsqueda.
[^...]	Ídem.

- Supongamos el contenido siguiente:

```
$ ls  
afic afic2 bfic bfic2 cfic cfic2 dfic dfic2  
afic1 afic3 bfic1 bfic3 cfic1 cfic3 dfic1 dfic3
```

Obtiene todos los ficheros que empiezan con a:

```
$ ls a*  
afic1 afic2 afic3
```

Todos los ficheros de cuatro caracteres que empiezan con a:

```
$ ls a???
```

Todos los ficheros de al menos tres caracteres y que empiezan con b:

```
$ ls b??*
```

Todos los ficheros que terminan con 1 o 2:

```
$ ls *[12]
afic1 afic2 bfic1 bfic2 cfic1 cfic2 dfic1 dfic2
```

Todos los ficheros que empiezan con las letras de a a c, que tienen al menos un segundo carácter antes de la terminación 1 o 2:

```
$ ls [a-c]?*[12]
afic1 afic2 bfic1 bfic2 cfic1 cfic2
```

Todos los ficheros que no terminan por 3:

```
$ ls *![3]
afic afic1 afic2 bfic bfic1 bfic2 cfic cfic1 cfic2 dfic
dfic1 dfic2
```

### **Interpretación por el shell**

El shell es el encargado de efectuar la sustitución de estos caracteres antes de pasar los parámetros a un comando. Así, en el momento de un \$ cp \* Documents, cp no recibe el carácter \*, sino la lista de todos los ficheros y directorios del directorio activo.

Los comodines pueden utilizarse dentro de todos los argumentos que representan ficheros o rutas. Así, el comando siguiente va a volver a copiar todos los ficheros README de todos los subdirectorios de Documents en la posición actual: \$ cp Documents/\*/README

## **d. Cierre de caracteres**

Se deben cerrar algunos caracteres especiales; por ejemplo, en caso de caracteres poco corrientes en un nombre de fichero.

- La **contrabarra** \ permite cerrar un carácter único. ls paga\ \*.xls va a listar todos los ficheros que contienen un espacio después de paga.
- Las **comillas** "..." permiten la interpretación de los caracteres especiales, de las variables, dentro de una cadena.
- Los **apóstrofes** '...' cierran todos los caracteres especiales en una cadena o un fichero.

# Buscar ficheros

## 1. Consideraciones generales

El comando **find** permite buscar ficheros dentro de la estructura del sistema de ficheros con la ayuda de criterios, y da la posibilidad de actuar sobre los resultados devueltos.

```
find ruta criterios opciones
```

Al ser recursivo el comando **find**, basta con indicar un directorio básico para que se desarrolle toda la estructura desde este directorio. La opción básica es **-print** (a menudo implícita en la mayoría de los Unix), que permite visualizar en pantalla los resultados.

```
seb@slyserver:~/Documents/slyunix> find
.
./logos-cuadrado.tif
./logos-cuadrado.eps
./Pagina 5.pdf
./logos-cuadrado-grande.jpg
./LOGOS
./sitio_2.jpg
./pub_planeta.pdf
./index_logon_inc.php
./logo-iceberg.eps
./flyer
./flyer/sly4.jpg
./flyer/flyerx4.sxd
./flyer/sly1.jpg
./flyer/sly2.jpg
./flyer/flyer.sxd
./flyer/sly3.jpg
./flyer/flyer.jpg
...
...
```

La visualización es relativa, ya que la ruta precisada es relativa. Si se hubiese especificado la ruta absoluta, la visualización habría sido absoluta.

## 2. Criterios de búsqueda

Los parámetros permiten definir los criterios de búsqueda. Si hay varios, puede combinarlos usando una Y lógica (criterio1 Y criterio2).

### a. -name

**-name** permite una selección por nombres de ficheros. Es posible utilizar los comodines ya vistos. Se coloca el criterio entre comillas. Aquí se visualiza la lista de todos los ficheros desde la ubicación actual y que empiezan por «fic».

```
$ find . -name "fic*" -print
./fic1
./fic2
./fic3
./fic4
```

## b. -type

`-type` permite una selección por tipo de ficheros. Ya sabe que, además de los vínculos, los directorios y los ficheros simples, están presentes otros tipos de ficheros.

Código	Tipo de fichero
B	Fichero especial en modo bloque
C	Fichero especial en modo carácter
D	Directorio (directory)
F	Fichero ordinario
L	Vínculo simbólico
P	Tubería con nombre (pipe)
S	Socket (Conexión de red)

Se visualizan todos los directorios cuyo nombre empieza por «re».

```
$ find . -name "re*" -tipo d -print  
./dir1  
./dir2
```

## c. -user y -group

`-user` y `-group` permiten una búsqueda sobre el propietario y el grupo de pertenencia de los ficheros. Es posible precisar el nombre (usuario, grupo) o la ID (UID, GID). El ejemplo siguiente busca todos los ficheros ordinarios que pertenecen a `seb` y al grupo `users`.

```
$ find . -tipo f -user seb -group users -print  
./fic1  
./fic3
```

## d. -size

`-size` permite precisar el tamaño de los ficheros buscados. Su sintaxis es particular, ya que trabaja por defecto en bloques si usted no precisa nada. Puede resultar sorprendente a veces; tanto más cuanto el bloque, que aquí tiene un tamaño de 512 bytes, es una unidad un poco virtual (con ciertos comandos, un bloque puede ser de 1 KB o más).

Los caracteres `b`, `c`, `w` o `k` pueden aparecer tras el valor colocado después del criterio.

Carácter	Significado
B	Por defecto, si no está precisado, es un bloque de 512 bytes.
C	Es un carácter en el sentido ASCII; por lo tanto, 1 byte.
W	Es una palabra (en el sentido antiguo) de 2 bytes.

K	1 KB (1024 bytes).
---	--------------------

El valor puede ir precedido de un + o un - que significan «más de» o «menos de». Sin esta indicación, el tamaño buscado debe corresponderse EXACTAMENTE.

- `-size 5`: busca los ficheros de un tamaño de 5 bloques (512 bytes por bloque, o sea, aquí 2560 bytes).
- `-size 152c`: busca los ficheros de un tamaño de 152 caracteres (bytes).
- `-size 10k`: busca los ficheros de un tamaño de 10 KB ( $10 \times 1024$  bytes = 10240 bytes).
- `-size +5000k`: los ficheros de más de 5000 KB.
- `-size -100k`: los ficheros de menos de 100 KB.

```
seb@slyserver:/var/log> find -size +100k
./zypper.log-20080227.bz2
./lastlog
./zypper.log-20080302.bz2
./wtmp
./zypper.log-20080226.bz2
./zypper.log
./messages
```

 Se puede utilizar el criterio de búsqueda **-empty** en sustitución de `-size 0`.

## e. **-atime**, **-mtime** y **-ctime**

- `-atime`: busca en la fecha del último acceso (access time). Un acceso puede ser la lectura del fichero, pero también el simple hecho de listarla de manera específica.
- `-mtime`: busca en la fecha de la última modificación (modification time). Se trata de la modificación del contenido.
- `-ctime`: busca en la fecha de modificación (change time, en realidad la fecha de última modificación del número de inodo).

 La fecha de modificación del fichero corresponde a la fecha en la cual se modificó por última vez la información relacionada con el inodo (ver capítulo Los discos y el sistema de ficheros): modificación del nombre, desplazamiento, cambio de derechos, del tamaño, etc.

Estos tres criterios sólo trabajan con días (periodos de 24 horas). 0 es el mismo día; 1, ayer; 2, antes de ayer, etc. El valor n colocado después del criterio corresponde, por lo tanto, a  $n \times 24$  horas. Este rango no es fijo, ya que «ayer» significa entre 24 y 48 horas...

Los signos + o - permiten precisar los términos «de más» y «de menos»:

- `-mtime 1`: ficheros modificados ayer (entre 24 y 48 horas).
- `-mtime -3`: ficheros modificados hace menos de tres días (72 horas).
- `-atime +4`: ficheros modificados hace más de cuatro días (más de 96 horas).

```
seb@slyserver:/var/log> find . -mtime -1
```

```
./kdm.log  
./vmware  
./vmware/vmware-serverd-0.log  
./vmware/vmware-serverd.log  
./mail.info  
./Xorg.0.log  
./lastlog  
./Xorg.0.log.old  
./warn  
...
```

 Eche un vistazo a los criterios **-newer**, **-anewer** y **-cnewer**, que toman un fichero como parámetro. En este caso, **find** busca los ficheros que son más recientes que el designado.

## f. -perm

**-perm** permite efectuar búsquedas en las autorizaciones de acceso (derechos, SUID, SGID, Sticky). Se deben precisar los derechos en base 8 (valor octal) y completos. El carácter **-** colocado antes del valor octal significa que los ficheros buscados deben tener al menos los permisos establecidos. En el ejemplo siguiente, se buscan los directorios sobre los cuales todo el mundo (user, group, others) tiene permiso de ejecución (permiso x, o sea 1).

```
seb@slyserver:/var/log> find -type d -perm -111  
. .  
./vmware  
./vmware/vmsd-xaction  
./cups
```

## g. -links y -inum

Aunque estos criterios se refieren a nociones más avanzadas del sistema de ficheros, es buena idea presentarlos ahora. Podrá volver a ellos en cuanto el capítulo Los discos y el sistema de ficheros le presente el funcionamiento interno de un sistema de ficheros.

La opción **-links** permite una búsqueda por nombre de hard links. Puede precisar los signos + o - (más de n vínculos y menos de n vínculos). Un fichero normal único posee 1 vínculo. Un directorio, 2 vínculos (la entrada en el catálogo al cual pertenece y en el punto). Para una búsqueda de vínculos simbólicos, habrá que utilizar la opción **-type l**.

```
$ find . -type f -links +2 -print  
./fic2  
./hardlink3_fic2  
./hardlink_fic2  
./hardlink2_fic2
```

**inum** permite una búsqueda por número de inodo. Es útil en el caso de una búsqueda de todos los vínculos que llevan un mismo número de inodo. El número de inodo puede verse mediante la opción **-idel** del comando **ls**.

```
seb@slyserver:/var/log> ls -i  
491891 acpid          491793 mail.info    491860 Xorg.0.log  
491791 boot.log        491794 mail.warn    490686 Xorg.0.log.old  
491729 boot.msg        492046 mcelog      492060 Xorg.1.log  
seb@slyserver:/var/log> find . -inum 491791 -print  
./boot.log
```

## 3. Comandos

Además de la opción `-print`, hay otras opciones que permiten efectuar una acción en los ficheros encontrados.

### a. -ls

El criterio muestra información detallada en los ficheros encontrados que corresponden al criterio, en lugar del simple nombre de fichero. La salida corresponde a un comando **ls** con los parámetros `d`, `i`, `l` y `s`(tamaño en bloques de 1 KB).

```
seb@slyserver:~> find -size +500000k -ls  
2342935 584388 -rw-r--r-- 1 seb users 597817344 feb 24  
11:52 ./eeexubuntu-7.10.3-desktop-i386.iso
```

### b. -exec

El criterio `-exec` va a ejecutar el comando colocado justo después con cada coincidencia encontrada. Es necesario hacer algunas observaciones:

- `-exec` debe ser obligatoriamente la última opción del comando **find**.
- El comando ejecutado por `-exec` debe terminarse con un «`>`». Este carácter especial debe escribirse con `\;` para que el shell no lo interprete.
- Con el fin de pasar como parámetro para el comando el fichero encontrado por `find`, hay que escribir `{}` (sustitución del fichero).

Ejemplo para borrar todos los ficheros que terminan por «.mp3»:

```
$ find . -type f -name "*.mp3" -exec rm -f {} \;
```

 El sistema no espera la finalización del comando **find** para ejecutar el comando **rm**. Lo ejecuta en cuanto **find** encuentra un fichero. Así, si el comando anterior le ha mostrado n ficheros antes de que usted pensara interrumpirlo, estos n ficheros ya han sido borrados.

### c. -ok

El criterio `-ok` es idéntico a la opción `-exec`, pero, para cada coincidencia, se le requiere una confirmación al usuario.

```
$ find . -inum 95 -ok rm -f {} \;  
< rm ... ./fic1 > (yes)? n  
< rm ... ./vínculo_fic1 > (yes)? y
```

## 4. Criterios AND / OR / NOT

Es posible combinar las opciones de criterio de selección. Si no se precisa ninguno, el Y lógico está implícito.

Criterio	Acción
<code>-a, -and</code>	AND, Y lógico, por defecto
<code>-o, -or</code>	OR, O lógico

Ejemplo con todos los ficheros que no contienen fic en su nombre, y todos los ficheros que no son ni normales ni directorios.

```
$ find . ! -name "*fic*" -print
.
./dir1
./lista
./mypass
./users
./lista2
./ls.txt
./pepito.tar.gz
./nohup.out
./lista_ls
./dir2
./seb1
./seb2
$ find . ! \(-type f -o -type d \) -ls
        409      0 lrwxrwxrwx  1 oracle    system          4 Ago 14 15:21
./vínculo_fic1 -> fic1
        634      0 lrwxrwxrwx  1 oracle    system          4 Ago 14 15:21
./vínculo_fic2 -> fic2
```

## 5. Encontrar ejecutables

### a. whereis

El comando **whereis** busca en las rutas de ficheros binarios, del manual y de las fuentes los ficheros que corresponden a los criterios facilitados.

```
$ whereis date
date: /bin/date /usr/share/man/man1/date.1.gz
/usr/share/man/man1p/date.1p.gz
```

Puede precisar algunos parámetros:

- **-b** únicamente para los binarios,
- **-m** únicamente para los manuales,
- **-s** únicamente para las fuentes.

Se buscan los ficheros por defecto en:

```
/{bin,sbin,etc}
/usr/{lib,bin,old,new,local,games,include,etc,src,man,sbin,X386,TeX,
g++-include}
/usr/local/{X386,TeX,X11,include,lib,man,etc,bin,games,emacs}
```

Por lo tanto, no se sorprenda de obtener lo siguiente:

```
$ whereis -b passwd
passwd: /usr/bin/passwd /etc/passwd /etc/passwd.old
/etc/passwd.YaST2save /etc/passwd.vipwKSNTgH /usr/bin/X11/passwd
```

### b. which

El comando **which** busca un comando en el PATH (ruta de los ejecutables) y le facilita el primero que encuentra:

```
$ which date  
/bin/date
```

En Linux hay comandos con el mismo nombre en varias rutas. Debe precisar el parámetro **-a** para que **which** siga su búsqueda. Sin embargo, si usted ejecuta el comando sin especificar una de esas rutas, se ejecutará el primero por defecto.

```
$ which -a passwd  
/usr/bin/passwd  
/usr/bin/X11/passwd
```

## 6. locate

El comando **locate** busca un fichero según el modelo dado en una base de datos de ficheros construida por el comando **updatedb**.

El comando **updatedb** recorre una serie de rutas en la cual va a efectuar un **find** y almacenar todos los resultados en una base indexada. Esto evita, por lo tanto, efectuar de nuevo un **find** para las búsquedas clásicas. En la práctica, basta con pasarle a **updatedb** la lista de las rutas o no incluir los ficheros en la base indexada.

contrab suele lanzar **updatedb** a diario. A veces se colocan los parámetros del comando en un fichero **/etc/sysconfig/locate**.

```
$ cat /etc/sysconfig/locate | grep -Ev "^(#|$)"  
RUN_UPDATEDB=yes  
RUN_UPDATEDB_AS=nobody  
UPDATEDB_NETPATHS=""  
UPDATEDB_PRUNEPATHS="/mnt /cdrom /tmp /usr/tmp /var/tmp /var/spool  
/proc /media /sys"  
UPDATEDB_NETUSER=""  
UPDATEDB_PRUNEFS=""
```

En este caso, se ejecuta el comando siguiente:

```
# updatedb --localuser=nobody --prunepaths=/mnt /cdrom /tmp /usr/tmp /var/tmp  
/var/spool /proc /media /sys
```

Si

usted ejecuta el comando de esta manera, puede que éste consuma todos los recursos de procesador de su máquina. Por eso, se ejecuta **updatedb** con una prioridad baja.

Se coloca la base de datos en **/var/lib/locatedb**.

```
$ locate pepito  
/opt/kde3/share/apps/ksgmltools2/docbook/xsl/html/auptotoc.xsl  
/opt/kde3/share/apps/ksgmltools2/docbook/xsl/params/autotoc.label  
.separator.xml  
/usr/share/gnome/help/gnome-doc-xslt/C/db2html-autotoc.xml  
/usr/share/xml/docbook/stylesheet/nwalsh/1.73.1/fo/autotoc.xsl  
/usr/share/xml/docbook/stylesheet/nwalsh/1.73.1/html/autotoc.xsl  
/usr/share/xml/docbook/stylesheet/nwalsh/1.73.1/params/autotoc.label  
.in.hyperlink.xml  
/usr/share/xml/docbook/stylesheet/nwalsh/1.73.1/params/autotoc.label  
.separator.xml  
/usr/share/xml/docbook/stylesheet/nwalsh/1.73.1/xhtml/autotoc.xsl
```

```
/usr/share/xml/gnome/xslt/docbook/html/db2html-autotoc.xsl  
/usr/src/linux-2.6.22.17-0.1/drivers/mtd/maps/omap-pepito-flash.c  
/usr/src/linux-2.6.22.17-0.1/drivers/mtd/nand/pepito.c  
/usr/src/linux-2.6.24.4/drivers/mtd/maps/omap-pepito-flash.c  
/usr/src/linux-2.6.24.4/drivers/mtd/nand/pepito.c
```

# El editor vi

## 1. Presentación

El editor de Unix por defecto se llama **vi** (*visual editor*). No es de los más ergonómicos en comparación con editores en modo gráfico, pero tiene la ventaja de estar disponible y de utilizar la misma sintaxis básica en todos los Unix. Cada Unix propone en general una sintaxis extendida más allá de la sintaxis básica. El editor vi bajo Linux se llama vim. Vim respeta toda la sintaxis de vi, pero no es recíproco. Vi es pequeño: ocupa poco espacio en disco, consume poca memoria. Algunas de las opciones disponibles sólo funcionan en vim, como por ejemplo la gestión de buffers de memoria múltiples.

`vi [opciones] Fichero [Fichero2 ...]`

Vi no tiene menús, tampoco interfaz gráfica, y no es intuitivo. Requiere conocerse de memoria un determinado número de atajos de teclado para poder utilizarlo. Si bien el aprendizaje es un poco difícil, una vez dominado vi se revela rápido y práctico, hasta tal punto que se va más rápido que con editores de texto gráficos.



El debate que enfrenta a los partidarios de emacs (o de otros editores) con los de vi no tiene razón de ser. Cualquier sistema Linux (y Unix) dispone pase lo que pase del editor vi, por lo que es ineludible. Si tiene la oportunidad, puede instalar el editor que le guste, aunque esto, en entornos empresariales, de servidores, etc., no siempre será posible.

## 2. Funcionamiento

Existen tres modos de funcionamiento:

- modo **comando**: las inserciones representan comandos. Se accede a ellos al presionar [Esc]. Cada tecla o combinación de teclas activa una acción (supresión de líneas, inserciones, mover, copiar, pegar, etc.).
- modo **inserción**: se trata de la inserción de texto clásico.
- modo **Línea de comandos**: una línea en la parte inferior de la pantalla permite insertar comandos especiales, validada con la tecla [Intro]. Se accede a ella al presionar, en modo comando, la tecla «`:`».

Cuando ejecuta vi, por defecto está en modo comando. Para empezar a introducir texto hay que teclear un comando de añadido o inserción, lo que enerva rápidamente cuando uno no conoce vi: a o i. Por lo tanto, el comando [Esc] i permite insertar texto. Para salir, puede pasar por el modo línea de comandos. Teclee [Esc]: y luego q y [Intro]. ¡Es sencillo!



Si ha usado un comando ya se encuentra en modo comando y no precisa pulsar la tecla Escape de nuevo para enviar el siguiente. Por ejemplo, la «`x`» suprime un carácter. Si ya está en modo comando, puede pulsar 10 veces la «`x`» sin darle a [Esc] entre cada pulsación.

Por supuesto, hay un punto de ironía. Nadie conoce todos los atajos; el propio autor de este libro no los recuerda todos. Por lo tanto, además de las guías en la prensa especializada, aquí tiene la lista de los comandos más utilizados en forma de tablas. Cuando el «`:`» precede al comando, éste se debe insertar en línea de comandos.

Si se pierde, en cualquier caso y pase lo que pase, al pulsar la tecla [Esc] vuelve siempre en modo comando.

### 3. Los comandos

#### a. La introducción de datos

Se deben efectuar las acciones siguientes en modo comando. Antes se debe pulsar [Esc]: [Esc] a, [Esc] i, etc.

Comando	Acción
<b>a</b>	Añadido después del carácter actual.
<b>A</b>	Añadido de texto al final de línea.
<b>i</b>	Inserción delante del carácter actual, como en un procesador de texto.
<b>I</b>	Inserción de texto al principio de línea.
<b>o</b>	Añadido de una línea debajo de la línea actual.
<b>O</b>	Inserción de una línea encima de la línea actual.

#### b. Salir y guardar

Recuerde que «:» significa que el comando se introduce en la línea de comandos: [Esc] :, inserción del comando, luego [Intro].

Comando	Acción
<b>ZZ</b>	Guarda el fichero y sale.
<b>:q!</b>	Sale sin guardar.
<b>:q</b>	Sale si no se modificó el fichero (en caso contrario, aparece un mensaje de error)
<b>:w</b>	Guarda el fichero. Puede precisar un nombre para seguir.
<b>:wq o :x</b>	Guarda y sale.
<b>1,10w fic</b>	Guarda las líneas de 1 a 10 en fic.

#### c. Desplazamiento

Quizá la idea le haga sonreír, pero existen todavía teclados sin teclas direccionales, así como programas e intérpretes de comando que no saben interpretarlos (ejemplo del shell ksh mal configurado en algunos Unix). En estas máquinas, se activa el modo de inserción de vi por defecto e, incluso bajo vi, es posible arreglárselas sin las teclas de dirección, en modo comando.

Comando	Acción
---------	--------

<b>h</b>	Ir a la izquierda.
<b>l</b> (l minúscula)	Ir a la derecha.
<b>k</b>	Ir hacia arriba.
<b>j</b>	Ir hacia abajo.
<b>0</b> (cero)	Principio de línea.
<b>:0</b>	Principio de fichero (primera línea).
<b>\$</b>	Fin de línea.
<b>:\$</b>	Fin de fichero (última línea).
<b>w</b>	Ir a la palabra siguiente.
<b>b</b>	Ir a la palabra anterior.
<b>f&lt;c&gt;</b>	Saltar al carácter <c> siguiente.
<b>Control+f</b>	Avanzar una pantalla (Forward).
<b>Control+b</b>	Retroceder una pantalla (Backward).
<b>G</b>	Última línea del fichero.
<b>&lt;n&gt;G</b>	Salta a la línea «n» (p. ej.: 10G va a la 10 <sup>a</sup> línea).
<b>:&lt;n&gt;</b>	Ídem ( :10 va a la 10 <sup>a</sup> línea).

#### d. La corrección

Comando	Acción
<b>x</b>	Borra el carácter bajo el cursor.
<b>X</b>	Borra el carácter delante el cursor.
<b>r&lt;c&gt;</b>	Sustituye el carácter bajo el cursor por el carácter <c>.
<b>dw</b>	Borra desde el cursor hasta el final de la palabra.
<b>d\$ o D</b>	Borra desde el cursor hasta el final de la línea.

<b>dO</b>	Borra desde el principio de línea hasta el cursor.
<b>df&lt;c&gt;</b>	Borra todo hasta el carácter <c>.
<b>dG</b>	Borra todo hasta la última línea, incluyendo la línea actual.
<b>d1G</b>	Borra todo hasta la primera línea, incluyendo la línea actual.
<b>dd</b>	Borra la línea actual.
<b>u</b>	Undo. Cancela la última acción.

Se pueden repetir estos comandos. 5dd suprime 5 líneas; 4dw, 4 palabras; 5x, 5 caracteres, etc.

## e. Búsqueda en el texto

A diferencia de un editor de texto clásico, vi puede buscar otras cosas además de palabras sencillas, y funciona con la ayuda de caracteres especiales y criterios. El comando de búsqueda es el carácter /. La búsqueda arranca en el carácter actual hasta el final del fichero. El carácter ? efectúa la búsqueda en sentido inverso. Primero se indica el criterio y luego se pulsa [Intro].

/echo

busca la cadena 'echo' a continuación en el fichero. Cuando se encuentra la cadena, el cursor se para en el primer carácter de esta cadena.

El comando n permite seguir la búsqueda en el sentido indicado al principio. El comando N efectúa la búsqueda en sentido inverso.

### Algunos criterios

- /[CcBb]ola: Cola, cola, Bola, bola.
- /[A-Z]e: todo lo que empieza con una mayúscula con una e en segunda posición.
- /[A-Za-Z0-9]: todo lo que empieza por una mayúscula, una minúscula o una cifra.
- /[^a-z]: intervalo negativo: todo lo que no empieza con una minúscula.
- /co.a: el punto sustituye un carácter: coma, cola, cota...
- /Au\*o: la estrella es un carácter de repetición, de 0 a n caracteres, Auo, Auto, Automoto...
- /\*.\* : la estrella delante del punto, una cadena cualquiera de tamaño variable; el "." representa un carácter.
- /[A-Z][A-Z]\*: repetición del motivo entre [] de 0 a n veces, búsqueda de una palabra que comporta al menos una mayúscula (al principio de palabra).
- /(^Auto: el ^ indica que la cadena buscada deberá estar al principio de línea.
- /Auto\$: el \$ indica que la cadena buscada deberá estar al final de línea.

## f. Comandos de sustitución

Para sustituir algo de texto, hay que colocarse al principio de la cadena a modificar, luego teclear uno de los comandos siguientes. A continuación, teclee simplemente su texto. El principio es sencillo. Es idéntico al de la supresión: se suprime la parte precisada y vi pasa al modo de inserción para la introducción de datos.

Comando	Acción
<b>cw</b>	Sustituye la palabra corriente.
<b>c\$</b>	Sustituye hasta el final de la línea.
<b>c0</b> (cero)	Sustituye hasta el principio de la línea.
<b>cf&lt;x&gt;</b>	Sustituye hasta el próximo carácter <x>.
<b>c/&lt;rech&gt;</b>	Sustituye hasta la próxima coincidencia de la cadena <rech>.

## g. Copiar - Pegar

El comando **v** permite una selección visual. Se subraya el texto y usted puede desplazar el cursor para seleccionar el texto. Luego utilice uno de los comandos siguientes:

- el comando «d» corresponde a cortar (mover);
- la c hace casi lo mismo, pero vi se queda en modo edición;
- para pegar el texto en el lugar elegido, se utiliza el comando p (detrás del carácter) o P (delante del carácter). Si se ha copiado una línea entera, se colocará debajo de la línea activa.

Las acciones siguientes son posibles en modo comando:

- Para copiar una línea: yy.
- Para copiar cinco líneas: 5yy.
- Para colocar las líneas copiadas en un sitio determinado: p.
- El editor vi dispone de 26 búferes de memoria para almacenar en ellos los datos que se pueden nombrar como se deseé. Para ello, se utiliza el ".
- Para copiar cinco palabras en la memoria m1: "m1y5w.
- Para pegar el contenido de la memoria m1 en un lugar determinado: "m1p.

## h. Sustitución

La sustitución permite reemplazar de manera automática varias coincidencias por otra cadena.

:[1<sup>a</sup> línea, última línea]s/Modelo/Sustitución/[gc]

Los números de líneas son opcionales. En este caso, la sustitución se hace en la línea corriente únicamente. Como sustitución de los números de línea, . determina la línea corriente, 1 la primera línea,\$ la última línea.

El modelo es un criterio de búsqueda presentado en estas últimas páginas. Sustitución es una cadena cualquiera que sustituirá el modelo.

Por defecto sólo se sustituye la primera coincidencia. La letra g indica que hay que sustituir todas las coincidencias. Con c, vi requiere una confirmación para cada una de las coincidencias. El siguiente ejemplo sustituye todas las coincidencias de Unix o unix en UNIX.

```
:1,$s/[Uu]nix/UNIX/g
```

## i. Otros

### **Edición avanzada**

Presentamos algunos comandos prácticos.

<b>:r fic</b>	Inserta el contenido de fic desde el lugar actual.
<b>:! cmd</b>	Ejecuta el comando. Pulse [Intro] para volver a vi.
<b>:r! cmd</b>	Se inserta el resultado del comando en el lugar actual.
<b>:e fic</b>	Carga el fichero fic para edición.
<b>:e#</b>	Conmuta entre los diferentes ficheros abiertos.

### **Comando set**

El comando **set** permite configurar el editor y acceder a sus opciones:

- set all: muestra el conjunto de las opciones posibles.
- set number (o nu) / nonumber (o nonu): muestra/suprime los números de líneas.
- set autoindent / noautoindent: se conserva la indentación durante un retorno a la línea.
- set showmatch / noshowmatch: durante la introducción de una llave o de una paréntesis de cierre, se visualiza el de apertura durante un instante muy corto, luego el editor vuelve al carácter corriente.
- set showmode / noshowmode: vi mostrará una línea de estado (INPUT MODE).
- set tabstop=x: define el número de caracteres para una tabulación.

Vim propone a menudo la coloración sintáctica. Detecta el tipo de fichero cargado y colorea las líneas, las contraseñas, etc. Puede desactivar la coloración:

- syntax off para desactivar;
- syntax on para activar.

# Redirecciones

## 1. Fundamentos

Las redirecciones son una de las más importantes posibilidades proporcionadas por el shell. Por redirección se entiende la posibilidad de redireccionar la visualización de la pantalla hacia un fichero, una impresora o cualquier otro periférico, los mensajes de errores hacia otro fichero, de sustituir la introducción vía teclado por el contenido de un fichero.

Cualquier flujo de datos en entrada o salida de comando pasa por un canal. Como sucede con el agua, es posible desviar el curso de los datos hacia otro destino o desde otra fuente.

Linux utiliza canales de entradas/salidas para leer y escribir sus datos. Por defecto, el canal de entrada es el teclado, y el canal de salida, la pantalla. Los errores, direccionados por defecto a la pantalla, son tratados como un canal especial.

Es posible redireccionar estos canales hacia ficheros o flujo de texto de manera transparente para los comandos Linux.

## 2. En salida

Se puede utilizar el carácter **>** para redireccionar la salida estándar (la que va normalmente en la pantalla). Luego se indica el nombre del fichero donde se colocarán los resultados de salida.

```
$ ls -l > resultado.txt
$ cat resultado.txt
total 1
-rw-r--r--    1 Administ ssh_user      0 Jul  4 12:04 PEPITO
-rw-r--r--    1 Administ ssh_user      0 Jul 25 15:13 resultado.txt
-rw-r--r--    1 Administ ssh_user    171 Jul 25 15:13 test.txt
```

Si

no existe, se creará el fichero. Si existe, se sobreescibirá su contenido, incluso si el comando tecleado no es correcto. **El shell empieza primero por crear el fichero y luego ejecuta el comando.**

 Es un aspecto importante de las redirecciones: se interpretan las redirecciones de derecha a izquierda, y se instalan las redirecciones ANTES de la ejecución de los comandos. Hay que crear el fichero antes de poder escribir en él. De ahí que, incluso si el comando es falso, se crea o se «chafa» el fichero...

Para añadir datos a continuación del fichero, o sea, sin sobreescribirlos, se utiliza la doble redirección **>>**. Se añade el resultado del comando al final del fichero.

```
$ ls -l > resultado.txt
$ date >> resultado.txt
$ cat resultado.txt
total 1
-rw-r--r--    1 Administ ssh_user      0 Jul  4 12:04 PEPITO
-rw-r--r--    1 Administ ssh_user      0 Jul 25 15:13 resultado.txt
-rw-r--r--    1 Administ ssh_user    171 Jul 25 15:13 test.txt
Thu Jul 25 15:20:12 2002
```

## 3. En entrada

Los comandos que esperan datos o parámetros desde el teclado pueden también recibirlas desde un fichero usando el carácter **<**. Un ejemplo con el comando **wc** (*word count*), que permite contar el número de líneas,

de palabras y de caracteres de un fichero:

```
$ wc < resultado.txt
 4      29     203
```

## 4. Documento en línea

La redirección << es particular. Permite la utilización de los documentos en línea. Encontrará a veces el término Herescript o Here Document. Esto permite la inserción de un texto hasta un punto dado y el envío de su resultado a un comando o un filtro. Se autorizan las redirecciones clásicas. Después del <<, ha de indicar una cadena que define el final de la inserción; por ejemplo, aquí 'end'.

```
$ tr "[a-z]" "[A-Z]" << end
> hola amigos
> esto es un ejemplo
> de herescript
> end
HOLA AMIGOS
ESTO ES UN EJEMPLO
DE HERESRIPT
```

## 5. Los canales estándares

Se puede considerar un canal como un fichero que posee su propio descriptor por defecto, y en el cual se puede leer o escribir.

- El canal de entrada estándar se llama **stdin** y lleva el descriptor 0.
- El canal de salida estándar se llama **stdout** y lleva el descriptor 1.
- El canal de error estándar se llama **stderr** y lleva el descriptor 2. Se puede redireccionar el canal de error hacia otro fichero.

```
$ rmdir directorio2
rmdir: `directorio2': No such file or directory
$ rmdir directorio2 2>error.log
$ cat error.log
rmdir: `directorio2': No such file or directory
```

Puede redireccionar los dos canales de salida a un único fichero poniéndolos en relación. Para ello, se utiliza el **>&**. También es importante saber en qué sentido el shell interpreta las redirecciones. El shell busca primero los caracteres <, >, >> al final de la línea, ya que las redirecciones suelen estar al final de comando. Por lo tanto, si usted quiere agrupar los dos canales de salida y de error en un mismo fichero, hay que proceder como a continuación.

```
$ ls -l > resultado.txt 2>&1
```

Se redirecciona la salida 2 hacia la salida 1; por lo tanto, los mensajes de error pasarán por la salida estándar. Luego se redirecciona el resultado de la salida estándar del comando **ls** hacia el fichero **resultado.txt**. Este fichero contendrá, por lo tanto, a la vez la salida estándar y la salida de error.

Puede utilizar los dos tipos de redirección a la vez:

```
$ wc < resultado.txt > cuenta.txt
$ cat cuenta.txt
 4      29     203
```

## 6. Apertura de canales

Existen tres canales estándares y se numeran de 0 a 2. Así, 0< equivale a < y 1> a >. El comando **exec** permite abrir otros siete canales numerados de 3 a 9. Por lo tanto, hay diez canales en total.

Usted puede e, incluso, debe considerar, en el marco de los procesos, sacar algunos resultados por el canal 3, otros por el 4, y así sucesivamente. Los canales están abiertos para la entrada y la salida.

```
$ exec 3>dump.log
$ ls -l >&3
$ cat dump.log
total 3952
-rw-r--r-- 1 seb users 167212 oct  9 09:27 battlestar_1280.jpg
drwxr-xr-x 2 seb users  4096 mar  4 08:51 bin
drwxr-xr-x 8 seb users  4096 mar  4 08:45 cxoffice
drwx----- 2 seb users  4096 mar 10 12:29 Desktop
drwx----- 13 seb users 4096 mar  6 11:49 Documents
-rw-r--r-- 1 seb users      0 mar 11 11:34 dump.log
-rw-r--r-- 1 seb users 3785296 dic 12 15:15 e3555_EeePC4G.pdf
drwxr-xr-x 3 seb users  4096 mar 10 11:16 Games
drwxr-xr-x 5 seb users  4096 mar 10 11:16 karchiver-3.4.2.b4
-rw-r--r-- 1 seb users   358 mar 11 08:51 lista
-rw-r--r-- 1 seb users   608 mar 11 09:14 tmpgrp
-rw-r--r-- 1 seb users  1555 mar 11 09:15 tmppwd
```

Todo lo que se escribirá en el canal 3 se escribirá en el fichero dump.log. Luego se puede cerrar el canal relacionándolo con un seudocanal (canal de cierre -).

```
$ exec 3>&-
```

## 7. Filtro: definición

Un **filtro** (o un comando filtro) es un programa que sabe escribir y leer datos por los canales estándares de entrada y salida. Modifica o trata si es preciso el contenido. **wc** es un filtro. Le presentamos algunos: **more**(muestra los datos página por página), **sort** (ordenación de los datos), **grep** (criterios de búsqueda).

## 8. Pipelines/tuberías

Las redirecciones de entrada/salida tales como las que acaba de ver permiten redireccionar los resultados hacia un fichero. Luego se puede «inyectar» este fichero en un filtro para extraerle otros resultados. Esto obliga a teclear dos líneas: una para la redirección hacia un fichero, otra para redireccionar este fichero hacia el filtro. Las **tuberías** o **pipes** permiten redireccionar el canal de salida de un comando hacia el canal de entrada de otro. El carácter que lo permite | está accesible con la combinación [AltGr] 1 de los teclados españoles.

```
$ ls -l > resultado.txt
$ wc < resultado.txt
```

se

convierte en

```
$ ls -l | wc
```

Es posible colocar varios | en una misma línea.

El

```
$ ls -l | wc | wc  
1 3 24
```

primer comando no tiene por qué ser un filtro. No es el caso más habitual. Lo importante es que se debe facilitar un resultado. Ídem para el último comando, que puede ser por ejemplo un comando de edición o impresión. Finalmente, el último comando puede ser objeto de una redirección en salida.

```
$ ls -l | wc > resultado.txt
```

# Filtros y herramientas

Un **filtro** (o un comando filtro) es un programa que sabe escribir y leer datos por los canales estándares de entrada y salida. Modifica o trata si es preciso el contenido. **wc** es un filtro. Las herramientas no siempre se comportan como filtros. Permiten un determinado número de acciones en ficheros y su contenido, como, por ejemplo, dar formato o imprimir.

## 1. Extracción de los nombres y rutas

El comando **basename** permite extraer el nombre del fichero en una ruta.

```
$ basename /tmp/seb/lista  
lista
```

comando **dirname** efectúa lo contrario, extrae la ruta.

```
$ dirname /tmp/seb/lista  
/tmp/seb
```

## 2. Búsqueda de líneas

Se trata de extraer líneas de un fichero según varios criterios. Para ello, usted dispone de tres comandos:**grep**, **egrep** y **fgrep**, que leen los datos o bien desde un fichero de entrada, o bien desde el canal de entrada estándar.

### a. grep

La sintaxis del comando **grep** es **grep [Opciones] modelo [Fichero1...]**.

El modelo se compone de criterios de búsqueda que se parecen mucho a los criterios ya expuestos para vi, por ejemplo. No hay que olvidar que se debe interpretar estos criterios con el comando **grep**, y no con el shell. Por lo tanto, hace falta cerrar todos los caracteres.

```
$ cat fic4  
Cerdo  
Ternera  
Buey  
rata  
Rata  
buey  
$ grep "^[bB]" fic4  
Buey  
buey
```

comando **grep** también puede tomar algunas opciones interesantes.

- **-v** efectúa la búsqueda inversa: se visualizan todas las líneas que no corresponden a los criterios.
- **-c** sólo devuelve el número de líneas encontradas, sin mostrarlas.
- **-i** no diferencia las mayúsculas de las minúsculas.
- **-n** indica el número de línea para cada línea encontrada.
- **-l** en el caso de ficheros múltiples, indica en qué fichero se ha encontrado la línea.

```
$ grep -i '^b' fic4 Buey buey
```

## b. egrep

El comando **egrep** extiende los criterios de búsqueda y puede aceptar un fichero de criterios en entrada. Equivale a un grep -E. Emplea como criterios expresiones regulares.

```
egrep -f fichero_criterio Fichero_búsqueda
```

Carácter especial	Significado
	O lógico, la expresión colocada antes o después debe desaparecer.
(...)	Agrupación de caracteres.
[...]	Un carácter tiene esta posición entre los indicados.
. (punto)	Cualquier carácter.
+	Repetición, el carácter colocado antes debe aparecer al menos una vez.
*	Repetición, el carácter colocado antes debe aparecer de cero a n veces.
?	El carácter colocado antes debe aparecer una vez como máximo.
{n}	El carácter colocado antes debe aparecer exactamente n veces.
{n,}	Aparece n veces o más.
{n,m}	Aparece entre n y m veces.
^	En principio de cadena.
\$	En final de cadena.

Únicamente «buenas tardes» y «buenas noches» que empezarán por una mayúscula o una minúscula si están solos en una línea:

```
^[bB]uenas(tardes|noches)$
```

Verificación muy escueta de la validez de una dirección IP:

```
echo $IP | egrep '([0-9]{1,3}\.){3}[0-9]{1,3}'
```

Esta línea se descompone de la manera siguiente:

- ([0-9]{1,3}\.){3}: www.xxx.yyy.
- [0-9]: un carácter entre 0 y 9

- {1,3}: repetido entre una y tres veces, por lo tanto: x, xx o xxx
- \.: seguido de un punto
- {3}: el conjunto tres veces

Luego [0-9]{1,3}: .zzz

- [0-9]: un carácter entre 0 y 9
- {1,3}: repetido entre una y tres veces

### c. fgrep

El comando **fgrep** es un grep simplificado y rápido (fast grep) y equivale a un grep -F. Acepta también un fichero de criterios de búsqueda, pero debe tratarse de criterios simples, sin caracteres especiales. Introduzca en el fichero de criterios líneas sencillas (texto y cifras), una búsqueda por línea. Fgrep va a buscar en un fichero meta o un flujo en entrada las líneas que corresponden a cada uno de los criterios.

## 3. sed

El aprendizaje de sed requeriría todo un libro. Sed es un editor de flujo (Stream Editor) que permite filtrar y transformar texto. Es un poco como un editor que permite modificar texto vía comandos de scripts, pero en un paso y sin edición interactiva. Utiliza un juego extendido de comandos procedente del editor ed. Su sintaxis básica es:

```
sed -e '<cmd>' arch
```

Para utilizar sed, hay que aprender y entender las expresiones regulares. El cuadro del comando **egrep** retoma la sintaxis básica de las expresiones. Cualquier libro sobre sed parte de estas expresiones y recíprocamente.

Sed se utiliza muy a menudo para sustituir valores por otros (sustitución) o suprimir líneas particulares (aunque se podría utilizar grep en este caso). La sintaxis básica de sustitución es la siguiente:

```
s/<antiguo>/<nuevo>/[g]
```

La g final permite realizar una sustitución sobre toda la línea en caso de haya varias coincidencias. Aquí tiene un ejemplo que sustituye NOMBRE por Pepito:

```
$ echo "Me llamo __NOMBRE__. ¿Te llamas __NOMBRE__?" | sed -e 's/__NOMBRE__/Pepito/'  
Me llamo Pepito. ¿Te llamas __NOMBRE?  
$ echo "Me llamo __NOMBRE__. ¿Te llamas __NOMBRE__?" | sed -e 's/__NOMBRE__/Pepito/g'  
Me llamo Pepito. ¿Te llamas Pepito?
```

Puede colocar un valor numérico en el campo nuevo para precisar, si la búsqueda comporta varios elementos agrupados por paréntesis, en qué elemento de los buscados debe trabajar. Aquí tenemos un ejemplo sencillo que añade asteriscos alrededor del nombre pepito:

```
$ echo pepito | sed -e "s/(pepito\)/**\1**/"  
**pepito**
```

Para suprimir todas las líneas vacías o que contienen únicamente espacios:

```
$ sed -e '/^ *$/d' fichero
```

## 4. Columnas y campos

El comando **cut** permite seleccionar columnas y campos en un fichero.

### a. Columnas

La sintaxis es la siguiente:

```
cut -cColumnas [ficheros...]
```

Una columna es la posición de un carácter en la línea. El primer carácter es la columna 1; el segundo, la columna 2, y así sucesivamente. Una línea de 80 caracteres dispone de 80 columnas. La numeración empieza a 1. Es el método ideal para ficheros planos y con formato fijo, donde cada campo empieza y termina con posiciones dadas.

El formato de selección de columna es el siguiente:

- una columna sola (p. ej. -c2 para la columna 2);
- un intervalo (p. ej. -c2-4 para las columnas 2, 3 y 4);
- una lista de columnas (p. ej. -c1,3,6 para las columnas 1, 3 y 6);
- los tres a la vez (p. ej. -c1-3,5,6,12-).

```
$ cat lista
Producto  precio  cantidades
ratón      30       15
disco      100      30
pantalla   300      20
teclado    45       30

$ cut -c1-5 lista
Produ
ratón
disco
panta
tecla

$ cut -c1-3,10-12,15
Prorx  cantidades
rat0    15
dis0    30
pan0    20
tec530
```

### b. Campos

El comando **cut** también permite seleccionar campos. Se deben delimitar estos campos por defecto por una tabulación, pero el parámetro **-d** permite seleccionar otro carácter (espacio, ;). La selección de los campos es idéntica a la de las columnas.

---

El carácter separador debe ser único. No es posible poner ni dos, ni tres, ni una cadena de separadores. Para eliminar los caracteres múltiples, utilice tr. Asimismo, el separador por defecto es la tabulación. Ahora bien, por defecto se sustituyen las tabulaciones con espacios dentro de los editores...

---

```
cut -dc -fCampos [ficheros...]
```

Le presentamos unos ejemplos. El fichero lista contiene campos separados por tabulaciones.

---

```
$ cat lista
Producto  precio  cantidades
ratón      30       15
duro       100      30
disco      100      30
pantalla   300      20
teclado    45       30
tarjeta    45       30

$ cut -f1 lista
Producto
ratón
duro
disco
pantalla
teclado
tarjeta

$ cut -f1,3 lista
Producto  cantidades
ratón      15
duro       30
disco      30
pantalla   20
teclado    30
tarjeta    30
```

 Observe que, si invierte el orden de los campos (-f3,1), no obtendrá el efecto deseado: los campos salen siempre en el sentido 1,3.

---

A continuación vemos cómo aislar los nombres de un grupo y sus identificadores respectivos:

---

```
$ cat /etc/group
seb@slyserver:~> cat /etc/group
at:!:25:
audio:x:17:
avahi:!:106:
beagleindex:!:107:
bin:x:1:daemon
cdrom:x:20:
console:x:21:
daemon:x:2:
dialout:x:16:seb,esteban,enrique,public
disk:x:6:

$ cut -d: -f1,3 /etc/group
at:25
audio:17
avahi:106
beagleindex:107
bin:1
cdrom:20
console:21
daemon:2
dialout:16
disk:6
```

 Si no hay delimitador (tabulación u otro) en una línea, **cut** muestra toda la línea.

---

## 5. Recuento de líneas

El comando **wc** (*word count*) permite contar las líneas, las palabras y los caracteres.

```
wc [-l] [-c] [-w] [-m] fic1
```

- **-l**: cuenta el número de líneas
- **-c**: cuenta el número de bytes
- **-w**: cuenta el número de palabras
- **-m**: cuenta el número de caracteres

```
$ wc lista  
12      48     234 lista
```

El

fichero lista contiene 12 líneas, 48 palabras y 234 caracteres.

## 6. Ordenación de líneas

El comando **sort** permite ordenar las líneas. Por defecto, la ordenación se hace sobre toda la tabla en orden creciente. La ordenación es posible a partir de uno o varios campos. El separador de campos por defecto es la tabulación o, al menos, un espacio. Si hay varios campos, el primero es el separador; los demás son caracteres del campo.

La sintaxis de sort ha evolucionado desde hace varios años y Linux ha aplicado un estándar. Además, ya no utiliza la antigua sintaxis basada en **+/-**. En su lugar, hay que utilizar el parámetro **-k**. La numeración de los campos empieza con 1.

```
sort [opciones] [-k pos1[,pos2]] [fic1...]
```

```
$ cat lista  
ratón    óptico  30    15  
duro     30giga  100   30  
duro     70giga  150   30  
disco    zip     12    30  
disco    blando  10    30  
pantalla 15    150   20  
pantalla 17    300   20  
pantalla 19    500   20  
teclado  105   45    30  
teclado  115   55    30  
tarjeta  sonido 45    30  
tarjeta  vídeo  145   30
```

A

continuación vemos cómo ordenar por orden alfabético a partir de la primera columna:

```
$ sort -k 1 lista  
disco    blando  10    30  
disco    zip     12    30  
duro     30giga  100   30  
duro     70giga  150   30  
pantalla 15    150   20  
pantalla 17    300   20  
pantalla 19    500   20  
ratón    óptico  30    15  
tarjeta  sonido 45    30
```

tarjeta	vídeo	145	30
teclado		105	45
teclado		115	55

## Algunos parámetros

Opción	Papel
-d	Dictionnary sort (ordenación de diccionario). Sólo toma como criterio de ordenación las letras, las cifras y los espacios.
-n	Ordenación numérica, ideal para las columnas de cifras.
-b	Ignora los espacios al principio del campo.
-f	No hay diferencias entre mayúsculas y minúsculas (conversión en minúsculas y luego ordenación).
-r	Reverse, ordenación en orden decreciente.
-tc	Nuevo delimitador de campo c.

Ejemplo: ordenación numérica a partir de los precios por productos en orden decreciente:

```
$ sort -n -r -k 3 lista
pantalla    19      500     20
pantalla    17      300     20
pantalla    15      150     20
duro        70giga   150     30
tarjeta     vídeo    145     30
duro        30giga   100     30
teclado     115     55      30
teclado     105     45      30
tarjeta     sonido   45      30
ratón       óptico   30      15
disco       zip      12      30
disco       blando   10      30
```

También es posible ejecutar la ordenación desde un determinado carácter de un campo. Para ello, debe especificar el «.pos»: -k1.3 empezará la ordenación a partir del tercer carácter del campo 1.

## 7. Supresión de las líneas repetidas

El comando **uniq** permite suprimir las líneas repetidas en flujos en entrada o ficheros ordenados. Por ejemplo, a continuación se muestra cómo sacar únicamente la lista de los GID realmente utilizados como grupo principal de usuarios:

```
$ cut -d: -f4 /etc/passwd | sort -n | uniq
0
1
2
7
8
12
13
```

14  
25  
49  
51  
62  
...

## 8. Unión de dos ficheros

### a. En los campos comunes

El comando **join** permite efectuar la unión de dos ficheros en función de un campo común. Se deben ordenar los dos ficheros en los campos especificados en la unión.

```
join [-tc] [-1 n] [-2 m] fic1 fic2
```

La opción **-t** indica los separadores, **-1** el campo del primer fichero y **-2** el campo del segundo fichero, en los cuales efectuar la unión. Observe que **join** gestiona mal los duplicados y puede que se detenga como consecuencia de ello.

 Es posible que el comando **join** no le proporcione el resultado esperado. La razón es que se detiene en cuanto no encuentra correspondencia entre dos líneas.

### b. Línea a línea

El comando **paste** agrupa n ficheros en uno. Para ello, concatena las líneas de cada uno de los ficheros en una sola línea: línea1 de fic1 con línea2 de fic2, línea3 de fic 3, y así sucesivamente. Es un poco lo contrario de cut. El separador por defecto es la tabulación, pero puede precisar un delimitador con **-d**.

```
$ cat fic1
lista_a
lista_b
lista_c

$ cat fic2
lista_a2
lista_b2
lista_c2

$ paste -d: fic1 fic2
lista_a:lista_a2
lista_b:lista_b2
lista_c:lista_c2
```

## 9. División de un fichero en partes

### a. Recortar

Aquí tenemos un comando muy práctico, **split**, que permite recortar un gran fichero en varios trozos con un tamaño determinado. Los sistemas de ficheros no son todos iguales frente al tamaño máximo de un fichero. Bajo Linux, el problema no es habitual, ya que un sistema de ficheros de tipo ext3 puede soportar ficheros de 1 TB (TB = TeraByte; 1 Terabyte = 1024 GB), o sea, el equivalente a unos 130 DVD de doble capa. Pero las bandas magnéticas, o en menor medida los discos removibles, no disponen de esta posibilidad.

Se suelen formatear las llaves USB o un disco externo con un sistema de ficheros de tipo VFAT, procedente del mundo de Microsoft. Este sistema de ficheros, que procede de DOS y luego de Windows 9x, garantiza

una compatibilidad entre todos los sistemas (Unix, Windows, Mac OS), ya que quien puede lo más puede lo menos. VFAT (o más bien FAT16 o FAT32) soporta únicamente ficheros de un tamaño máximo de 4 GB. Una imagen ISO de DVD o una carpeta de copia de seguridad no puede entrar en ello en un solo bloque. Por lo tanto, hace falta dividir el fichero en varias partes.

```
split [-l n] [-b n[bkm]] [fichero [prefijo]]
```

El comando puede funcionar según dos modos:

- recorte por líneas con **-l**: los ficheros en salida tendrán todos n líneas de texto (salvo el último si se da el caso);
- recorte a tamaño fijo con **-b**: los ficheros tendrán todos un tamaño fijo de n bytes. El sufijo b indica un tamaño de n bloques (512 bytes), k indica n kB (1024 bytes) y m indica n MB (1024 kB).

Como cualquier filtro, **split** puede coger un flujo de entrada, como es el caso si no se precisa ningún fichero, o si hay un guión. Un prefijo define el nombre de los ficheros en salida. Aquí tenemos un fichero de 1 GB a cortar en partes de 150 MB. El prefijo es **fic**. Cada fichero en salida se llama **ficaa**, **ficab**, **ficac**, **ficad**, y así sucesivamente.

```
$ ls -l granfichero
-rw-r--r-- 1 seb users 1073741824 mar 12 19:47 granfichero
$ split -b 150m granfichero fic
$ ls -l fic*
-rw-r--r-- 1 seb users 157286400 mar 12 20:15 ficaa
-rw-r--r-- 1 seb users 157286400 mar 12 20:15 ficab
-rw-r--r-- 1 seb users 157286400 mar 12 20:15 ficac
-rw-r--r-- 1 seb users 157286400 mar 12 20:16 ficad
-rw-r--r-- 1 seb users 157286400 mar 12 20:16 ficae
-rw-r--r-- 1 seb users 157286400 mar 12 20:16 ficaf
-rw-r--r-- 1 seb users 130023424 mar 12 20:16 ficag
```

## b. Reconstruir

Una línea basta para reconstruir un fichero dividido con la ayuda de las redirecciones:

```
$ cat fic* > newfic
$ ls -l newfic
-rw-r--r-- 1 seb users 1073741824 mar 12 20:47 newfic
```

# 10. Sustitución de caracteres

## a. Lista de caracteres

El comando **tr** permite sustituir unos caracteres con otros y sólo acepta datos que provengan del canal de entrada estándar, no de los ficheros.

```
tr [opciones] original destino
```

El original y el destino representan uno o varios caracteres. Se sustituyen los caracteres originales con los de destino en el orden indicado. Los corchetes permiten definir intervalos.

Por ejemplo, para sustituir la o por la e y la i por la a:

```
$ cat lista | tr "oi" "ea"
Predut    ejete   precie   cantadades
raten    éptaque 30      15
dure     30gaga   100     30
```

dure	70gaga	150	30
dasce	zap	12	30
dasce	blande	10	30
pantalla	15	150	20
pantalla	17	300	20
pantalla	19	500	20
teclade	105	45	30
teclade	115	55	30
tarjeta	senade	45	30
tarjeta	vadée	145	30

Con este comando, puede convertir una cadena en mayúsculas o en minúsculas.

```
$ cat lista | tr "[a-z]" "[A-Z]"
PRODUCTO  OBJETO  PRECIO  CANTIDADES
RATÓN    ÓPTICO   30      15
DURO     30GIGA   100     30
DURO     70GIGA   150     30
DISCO    ZIP      12      30
DISCO    BLANDO   10      30
PANTALLA 15      150     20
PANTALLA 17      300     20
PANTALLA 19      500     20
TECLADO  105     45      30
TECLADO  115     55      30
TARJETA SONIDO  45      30
TARJETA VÍDEO   145     30
```

### **Suprimir las repeticiones**

Sobre todo, `tr` admite dos parámetros, `-s` (squeeze) y `-d` (delete), que permiten suprimir caracteres, duplicados o no. Es perfecto en el caso de separadores múltiples. A continuación damos un ejemplo práctico en el cual se busca aislar la dirección IP de una máquina.

```
$ /sbin/ifconfig eth0
eth0      Vínculo encap:Ethernet  HWaddr 00:13:D3:D7:A4:6C
          inet adr:10.9.238.170  Bcast:10.9.239.255  msk:255.255.252.0
                  UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
                  RX packets:15054381 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:4991811 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 lg file transmission:1000
                  RX bytes:4157389034 (3964.7 Mb)  TX bytes:374974072 (357.6 Mb)
                  Interrupción:22 Dirección básica:0xcc00
```

Sólo le interesa la línea que contiene `inet`:

```
$ /sbin/ifconfig eth0| grep "inet "
          inet dir:10.9.238.170  Bcast:10.9.239.255  Máscara:255.255.252.0
```

Para aislar la dirección IP colocada después de «`inet dir:`» el separador `<:>` puede parecer interesante, pero en este caso un `cut` nos devolvería «`10.9.238.170 Bcast`», lo que no conviene. La artimaña consiste en sustituir todos los espacios por un solo `<:>`. El parámetro `-s` sustituye una cadena de n caracteres idénticos por uno solo. Si no se precisa, es el mismo carácter; en caso contrario, se trata de un carácter de sustitución determinado.

```
$ /sbin/ifconfig eth0| grep "inet " | tr -s " " ":" 
:inet:dir:10.9.238.170:Bcast:10.9.239.255:Máscara:255.255.252.0
```

Ya sólo falta contar: la dirección IP está en cuarta posición (el primer campo antes del primer «:» está vacío).

```
$ /sbin/ifconfig eth0| grep "inet " | tr -s " \" ":" | cut -d: -f4  
10.9.238.170
```

## b. Tabulaciones y espacios

La mayoría de los editores sustituyen las tabulaciones por espacios. Ahora bien, algunos comandos esperan a obtener tabulaciones como delimitadores de campos (es el caso de **cut**). Si no puede apañarse con **tr**, tiene a su disposición dos comandos para este caso específico.

El comando **expand** convierte las tabulaciones en espacios. El comando **unexpand** convierte los espacios en tabulaciones. O sea, el fichero lista según el modelo anterior, en el cual se separan las columnas por espacios en lugar de tabulaciones. En el primer caso, el resultado no es el esperado para nada. El comando **cut** intenta sacar el tercer campo de un fichero tabulado. Como no hay tabulaciones, muestra toda la línea.

```
$ cut -f1 lista  
Producto      objeto      precio      cantidades  
ratón          óptico      30          15  
duro           30giga     100         30  
duro           70giga     150         30  
disco          zip         12          30  
disco          blando     10          30  
...  
El
```

comando **unexpand** con el parámetro **-a** sustituye todas las secuencias de al menos dos espacios por el número necesario de tabulaciones. Esta vez el resultado es correcto.

```
$ unexpand -a lista | cut -f1  
Producto  
ratón  
duro  
duro  
disco  
disco  
...  
El
```

# 11. Visualización de texto

## a. En pantalla completa

Nada impide desviar cualquier flujo para visualizarlo en la pantalla o por impresora. Aquí presentamos algunos comandos.

- página por página: **pg**, **more**, **less**
- en bloque: **cat**
- al revés: **tac**
- en dump hexadecimal: **hexdump**
- creación de un banner: **banner**
- formateo para impresión: **pr**
- numerar las líneas: **cat -n** o **nl**

## b. El principio de un fichero

Para ver el principio del contenido de un fichero, utilice el comando **head**.

```
head [-c nbyte] [-n nlineas] [fich...]
```

El parámetro **-c** permite precisar el número de bytes de encabezamiento a visualizar. Por defecto se visualizan diez líneas. El parámetro **-n** permite indicar el número de líneas a visualizar. Es posible indicar directamente el número de líneas:

```
head [-nlineas] [fich...]
```

```
$ head -3 lista
Producto objeto    precio  cantidades
ratón    óptico     30      15
duro     30giga    100     30
```

## c. Fin y modo de espera de fichero

Para ver las últimas líneas de un fichero, utilice el comando **tail**.

```
tail [+/-valor[b/c]] [-f] [fich...]
```

Al igual que para **head**, por defecto se visualizan las diez últimas líneas. El valor **-numlineas** permite modificar este estado. Use **c** para indicar un número de caracteres. Una **b** indica un número de bloques (512 bytes por bloque).

Un **+ invierte el orden del comando y se convierte en un **head** (**tail +10 <=> head -n 10**).**

Finalmente, la opción **-f** deja el fichero abierto. Si se inserta más información en él (por ejemplo, un fichero de registros), se visualizará su contenido en tiempo real en la pantalla hasta que el usuario lo interrumpe de manera voluntaria ([Ctrl] **C**).

```
$ tail -5 lista
pantalla 19      500      20
teclado   105     45       30
teclado   115     55       30
tarjeta   sonido  45       30
tarjeta   vídeo   145     30

$ tail -10c lista
eo        145     30
```

## d. Dar formato a una salida

El comando **column** permite dar formato de tabla a la salida de un comando. La opción **-t** determina cuántas columnas se mostrarán en la salida y añade espacios para alinearlas. La opción **-s** permite indicar cuál es el separador.

```
$ column -s: -t /etc/group
root          x  0
daemon        x  1
bin           x  2
sys           x  3
adm           x  4  seb
tty           x  5
disk          x  6
lp            x  7
```

## 12. Duplicación del canal de salida estándar

En algunos casos, como por ejemplo en la generación de ficheros de registros, puede ser necesario colocar el resultado de un comando en un fichero y a la vez filtrar este mismo resultado con otro comando. Para ello, utilice el comando **tee**, que permite duplicar el flujo de datos. Este comando lee el flujo de datos que proviene de otro comando por el canal de entrada, lo escribe en un fichero y restituye este flujo de forma idéntica por el canal de salida. Por defecto, el fichero generado sobrescribe el antiguo si existe.

```
tee [-a] nombre_fic
```

El parámetro **-a** significa **append**. En este caso, no se sobrescribe el fichero, sino que se inserta la información al final. Por ejemplo, supongamos que quiere obtener en un fichero la lista de los nombres de usuario y que al mismo tiempo se visualice su número en la pantalla.

```
$ cat /etc/passwd | cut -d: -f1 | tee users | wc -l
      65
$ cat users
root
nobody
nobodyV
daemon
bin
uucp
uucpa
auth
cron
lp
tcb
...
...
```

## 13. Comparación de ficheros

Los dos comandos que permiten comparar el contenido de dos ficheros, o de un fichero y de un flujo, son los comandos **diff** y **cmp**.

### a. diff

El comando **diff** indica las modificaciones que hay que aportar a los dos ficheros en entrada para que su contenido sea idéntico.

```
diff [-b] [-e] fic1 fic2
```

La opción **-b** permite ignorar los espacios (blank), y la opción **-e** permite generar un script ed (no lo utilizaremos). Este comando devuelve tres tipos de mensajes:

- APPEND: línea1 a línea3, línea4, ex 5 a 6,8 significa: en la línea 5 de fic1 hay que insertar las líneas 6 a 8 de fic2 para que sus contenidos sean idénticos.
- DELETE: línea1, línea2 d línea3, ex 7,9 d 6 significa: se deben suprimir las líneas 7 a 9 de fic1, no existen detrás de la línea 6 de fic2.
- CHANGE: línea1, línea2 c línea3, línea4, ex 8,12 c 9,13 significa: se debe intercambiar las líneas 8 a 12 de fic1 contra las líneas 9 a 13 de fic2.

En todos los casos, el signo «<>» indica las líneas de fic1 concernientes y el signo «>>» las líneas de fic2 concernientes.

```
$ cat lista
```

```

Producto objeto precio cantidades
ratón óptico 30 15
duro 30giga 100 30
duro 70giga 150 30
disco zip 12 30
disco blando 10 30
pantalla 15 150 20
pantalla 17 300 20
pantalla 19 500 20
teclado 105 45 30
teclado 115 55 30
tarjeta sonido 45 30
tarjeta vídeo 145 30

```

```

$ cat lista2
Producto objeto precio cantidades
ratón botones 30 15
duro 30giga 100 30
duro 70giga 150 30
disco zip 12 30
disco blando 10 30
pantalla 15 150 20
pantalla 17 300 20
pantalla 19 500 20
pantalla 21 500 20
teclado 105 45 30
teclado 115 55 30

```

fichero lista es el original. En lista2 se ha modificado la segunda línea, se ha añadido una línea pantalla y se han suprimido las últimas dos líneas.

```

$ diff lista lista2
2c2
< ratón óptico 30 15
---
> ratón botones 30 15
9a10
> pantalla 21 500 20
12,13d12
< tarjeta sonido 45 30
< tarjeta vídeo 145 30

```

2c2: se deben intercambiar las líneas 2 de lista y lista2 (deben concordar o en óptico o en botones).

- 9a10: después de la línea 9 de lista (pantalla 19), hay que añadir la línea 10 (pantalla 21) de lista2.
- 12,13d12: se deben suprimir las líneas 12 y 13 de lista (tarjeta de sonido y vídeo), ya que no existen después de la línea 12 de lista2.

## b. cmp

El comando **cmp** compara los ficheros carácter por carácter. Por defecto, el comando se para en cuanto encuentra la primera diferencia e indica la posición.

```
cmp [-l] [-s] fic1 fic2
```

El parámetro **-l** detalla todas las diferencias en tres columnas. La primera columna representa el número de carácter; la segunda, el valor octal ASCII del carácter correspondiente de fic1, y la tercera, el valor octal ASCII del carácter correspondiente de fic2.

La opción **-s** devuelve únicamente el código de error (no visible), al que se puede acceder por echo \$?.

```
$ cmp lista lista2
lista lista2 differ: char 38, line 2
$ cmp -l lista lista2
 38 157 142
 39 160 157
 40 164 165
 41 151 164
 42 161 157
 43 165 156
 44 145 163
182 143 145
183 154 143
...
```

## 14. Plazo de espera

El comando **sleep** permite esperar el número de segundos indicados. El script se interrumpe durante este tiempo: el número de segundos y un entero comprendido entre 0 y cuatro mil millones (136 años).

```
$ sleep 10
```

# Los procesos

## 1. Definición y entorno

Un **proceso** representa un programa en curso de ejecución y, al mismo tiempo, todo su entorno de ejecución (memoria, estado, identificación, propietario, padre...).

Los datos de identificación de un proceso son:

- **Un número de proceso único PID** (Process ID): se numera cada proceso Unix con el fin de poder diferenciarlo de los otros. El primer proceso iniciado por el sistema es 1, y se trata de un proceso llamado generalmente init. Se utiliza el PID cuando se trabaja con un proceso. Iniciar 10 veces el mismo programa (mismo nombre) produce 10 PID diferentes.
- **Un número de proceso padre PID** (Parent Process ID): cada proceso puede iniciar otros procesos, sus procesos hijos (child process). Cada proceso hijo debe contener, entre toda su información, el PID del proceso padre que lo inició. Todos los procesos tienen un PPID salvo el proceso 0, que es un seudoproceso que representa el inicio del sistema (crea el 1 init).
- **Un número de usuario y uno de grupo**: corresponde al UID y al GID de la cuenta de usuario que inicia el proceso. El sistema lo utiliza para determinar, a través de la cuenta, los permisos que el proceso tiene para acceder a los recursos. Los procesos hijos heredan ambas cuentas. En algunos casos (que veremos más adelante) se puede modificar este comportamiento.
- **Duración y prioridad del proceso**: la duración del proceso corresponde al tiempo de ejecución consumido desde la última invocación. En un entorno multitarea, el tiempo de procesador se comparte entre los procesos y no todos tienen la misma prioridad. Los procesos de más alta prioridad son ejecutados primero. Cuando un proceso está inactivo, su prioridad aumenta con el fin de tener la oportunidad de ser ejecutado. Cuando está activo, su prioridad baja con el fin de dejar paso a otro. Es el planificador de tareas del sistema el que gestiona las prioridades y los tiempos de ejecución.
- **Directorio de trabajo activo**: tras su inicio, se configura el directorio actual del proceso (desde el cual se inició). Este directorio servirá de base para las rutas relativas.
- **Ficheros abiertos**: tabla de los descriptores de ficheros abiertos. Por defecto al principio sólo hay tres presentes: 0, 1 y 2 (los canales estándares). Con cada apertura de fichero o de nuevo canal, la tabla se rellena. Al cierre del proceso, se cierran los descriptores (en principio).
- Puede encontrar más información, como el tamaño de la memoria asignada, la fecha de inicio del proceso, el terminal de atribución, el estado del proceso, los UID efectivo y real, así como los GID efectivo y real.

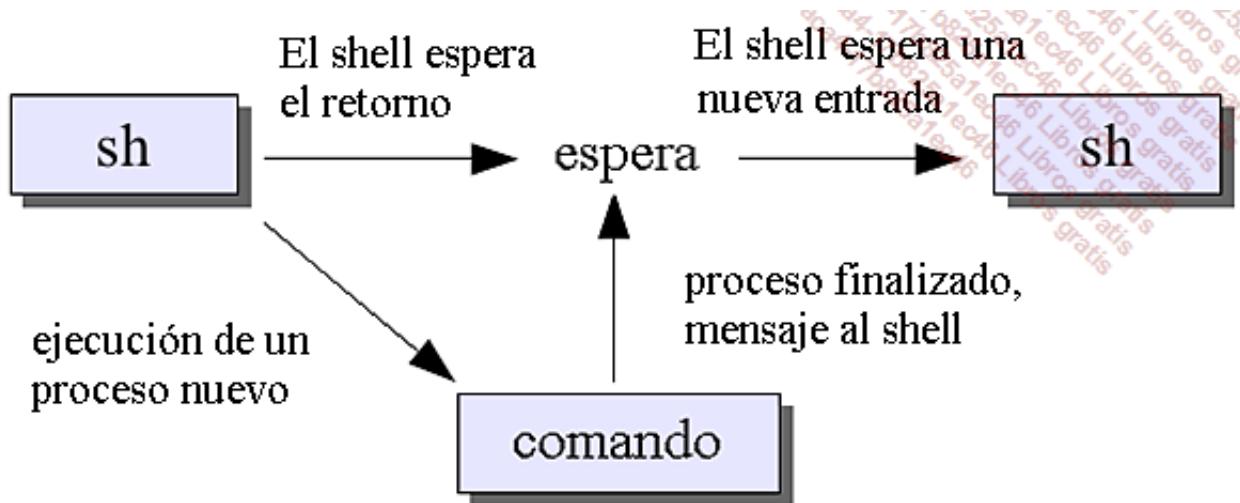
## 2. Estados de un proceso

Durante su vida (tiempo entre el inicio y la salida) un proceso puede pasar por diversos estados o **process state**:

- ejecución en modo usuario (**user mode**);
- ejecución en modo núcleo (**kernel mode**);
- en espera E/S (**waiting**);
- dormido (**sleeping**);
- listo para la ejecución (**runnable**);

- dormido en el swap (**memoria virtual**);
- nuevo proceso;
- fin de proceso (**zombie**).

### 3. Ejecución en segundo plano



Según lo visto anteriormente, dado que el entorno de trabajo es multitarea, hay un determinado número de procesos que se ejecutan ya en su máquina sin que usted lo vea. Del mismo modo, el shell que utiliza es en sí misma un proceso. Cuando inserta un comando, el shell crea un nuevo proceso para ejecutarlo, este proceso se convierte en un proceso hijo del shell. Una vez iniciado, hay que esperar al final de su ejecución para iniciar el siguiente (excepto utilizando «;» para encadenar los comandos).

Nada impide al shell esperar el mensaje del proceso terminado para dejar paso: de hecho, una vez iniciado el comando, el shell puede autorizar la inserción de un nuevo comando sin esperar el final de la ejecución del comando anterior. Para ello, basta insertar, después de haber tecleado el comando, el **ampersand <&>**. En este caso, el shell y el comando iniciado funcionarán en paralelo.

```
$ ls -R / > ls.txt 2>/dev/null &
[1] 21976
$ [1] Done
$ ls
f1c1      f1c3      lista      ls.txt      dir1      users
f1c2      f1c4      lista2     mypass     pepito.tar.gz
```

Justo después de la inserción, aparece una cifra. Hay que recordarlo, ya que se trata del PID del nuevo proceso iniciado en segundo plano. Después de otra inserción, un línea Done indica que el tratamiento se da por finalizado. El valor [1] es propio de un shell particular (ksh).

Algunas observaciones en cuanto al uso del inicio en segundo plano:

- El proceso iniciado en segundo plano no debería esperar ser otro shell, dado que se produciría confusión entre este comando y el propio shell.
- El proceso iniciado no debería mostrar resultados en la pantalla porque entrarían en conflicto con los del shell (por ejemplo, aparición de una línea en medio de una inserción).
- Finalmente, cuando se sale del shell, se sale también de todos sus hijos: en este caso, no abandone el shell en mitad de un procesamiento importante.

## 4. Background, foreground, jobs

Puede retomar el control del shell si ha iniciado un proceso en segundo plano. Puede pararlo de manera temporal tecleando [Ctrl] Z:

```
$ sleep 100
<CTRL+Z>
[1]+  Stopped                  sleep 100
```

Se ha parado el proceso: se ha suspendido su ejecución hasta que vuelva a ponerlo en primer plano con el comando **fg** (*foreground*):

```
$ fg
sleep 100
```

Cuando ejecuta un comando, obtiene un número entre llaves: es el número de job. Puede obtener una lista de todas las tareas con el comando **jobs**.

```
$ jobs
[1]-  Stopped                  sleep 100
[2]+  Stopped                  sleep 100
```

Los comandos **bg** y **fg** permiten actuar en estos jobs tomando como parámetro su número. Se ejecuta el comando **bg** en un job parado para iniciar lo nuevo en segundo plano. Se vuelve a iniciar el job 2 en segundo plano:

```
$ bg 2
[2]+ sleep 100 &
$
[2]+ Done                  sleep 100
```

## 5. Lista de los procesos

El comando **ps** (*process status*) permite obtener información sobre los procesos en curso. Si se ejecuta solo, visualiza únicamente los procesos en curso iniciados por el usuario y desde la consola actual.

```
$ ps
 PID TTY      TIME CMD
 4334 pts/1    00:00:00 bash
 5017 pts/1    00:00:00 ps
```

Para obtener más información, utilice el parámetro **-f**.

```
ps -f
UID      PID  PPID  C STIME TTY      TIME CMD
seb     4334  24449  0 09:46 pts/1    00:00:00 /bin/bash
seb     5024   4334  0 10:51 pts/1    00:00:00 ps -f
```

parámetro **-e** da información sobre todos los procesos en curso.

```
$ ps -ef
UID      PID  PPID  C STIME TTY      TIME CMD
...
seb     26431      1  0 Mar04 ?      00:00:30 kded [kdeinit]
seb     26436  26322  0 Mar04 ?      00:00:03 kwrapper ksmserver
seb     26438      1  0 Mar04 ?      00:00:00 ksmserver [kdeinit]
```

```

seb    26439 26424  0 Mar04 ?          00:00:50 kwin [kdeinit]
seb    26441     1  0 Mar04 ?          00:00:28 kdesktop [kdeinit]
seb    26443     1  0 Mar04 ?          00:03:40 kicker [kdeinit]
seb    26453     1  0 Mar04 ?          00:00:00 kerry [kdeinit]
seb    26454 26424  0 Mar04 ?          00:00:01 evolution
seb    26465 26424  0 Mar04 ?          00:00:11 kde-window-decorator
seb    26467     1  0 Mar04 ?          00:00:02 gconfd-2 12
seb    26474     1  0 Mar04 ?          00:00:01 knotify [kdeinit]
seb    26485     1  0 Mar04 ?          00:03:06 beagled
...

```

parámetro **-u** permite precisar una lista de uno o varios usuarios separados por una coma. El parámetro **-g** efectúa lo mismo pero para los grupos, **-t** para los terminales y **-p** para unos PID determinados.

```

$ ps -u root
PID TTY      TIME CMD
 1 ?        00:00:05 init
 2 ?        00:00:00 kthreadd
 3 ?        00:00:00 migration/0
 4 ?        00:00:09 ksoftirqd/0
 5 ?        00:00:23 events/0
 6 ?        00:00:00 khelper
25 ?        00:00:00 kblockd/0
26 ?        00:00:00 kacpid
27 ?        00:00:00 kacpi_notify
130 ?       00:00:00 cqueue/0
131 ?       00:00:00 kseriod
156 ?       00:00:22 kswapd0
157 ?       00:00:00 aio/0...

```

Finalmente el parámetro **-l** ofrece más información técnica.

```

A
$ ps -l
F S   UID   PID   PPID   C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S 1000  4704 24449  0 75    0 - 1213 wait    pts/3    00:00:00 bash

```

continuación presentamos el detalle de algunas columnas.

Columna	Definición
<b>UID</b>	User ID, nombre del usuario.
<b>PID</b>	Process ID, número del proceso.
<b>PPID</b>	Parent Process ID, número del proceso padre.
<b>C</b>	Factor de prioridad, cuanto más grande es el valor, más elevada es la prioridad.
<b>STIME</b>	Hora de inicio del proceso.
<b>TTY</b>	Nombre del terminal desde el cual se ejecutó el proceso.
<b>TIME</b>	Duración de tratamiento del proceso.
<b>CMD</b>	Comando ejecutado.

<b>F</b>	Banderas del proceso (sale del ámbito de este libro).
<b>S</b>	Estado del proceso S (sleeping) R (running) Z (zombie).
<b>PRI</b>	Prioridad del proceso.
<b>NI</b>	Nice, incremento para el scheduler.

## 6. Parada de un proceso/señales

Es posible detener un proceso en primer plano con la combinación de teclas [Ctrl] **Z**. Cuando un proceso se ejecuta en segundo plano, no lo puede parar cualquier combinación de teclas, al menos que se utilice el gestor de jobs con fg y bg. Puede resultar necesario mandarle señales especiales a ese proceso en segundo plano. Para ello, hay que emplear el comando **kill**. A diferencia de lo que parece indicar su nombre, el papel de este comando no es obligatoriamente destruir o terminar un proceso (recalcitrante o no), sino mandar señales a los procesos.

```
kill [-l] -Num_señal PID [PID2...]
```

La **señal** es uno de los medios de comunicación entre los procesos. Cuando se manda una señal a un proceso, éste debe interceptarla y reaccionar en consecuencia. Se pueden ignorar algunas señales, pero otras no. Según los Unix, se dispone de un número más o menos importante de señales. Se numeran y nombran las señales, pero cuidado: si bien los nombres suelen ser comunes de un Unix a otro, los números no lo son obligatoriamente. La opción **-l** permite obtener la lista de las señales.

Señal	Papel
1 (SIGHUP)	El padre manda Hang Up a todos sus hijos cuando termina.
2 (SIGINT)	Interrupción del proceso pedido (tecla [Supr], [Ctrl] <b>C</b> ).
3 (SIGQUIT)	Ídem SIGINT, pero con generación de un Core Dump (fichero de depuración).
9 (SIGKILL)	Señal que no se puede ignorar, fuerza el proceso a terminar de manera «expeditiva».
15 (SIGTERM)	Señal mandada por defecto por el comando <b>kill</b> . Pide al proceso terminar con normalidad.

```
$ sleep 100&
[1] 5187
$ kill 5187
$
[1]+  Completado          sleep 100
$ sleep 100&
[1] 5194
$ kill -9 5194
[1]+  Proceso parado      sleep 100
```

## 7. nohup

Cuando se sale del shell (exit, [Ctrl] D...) se manda la señal 1 SIGHUP a los hijos para que terminen ellos también. Cuando se inicia un proceso largo en segundo plano y el usuario quiere salir del shell, entonces se para este proceso y habrá que volver a empezarlo. La manera de evitar esto es iniciar el proceso con el comando **nohup**. En este caso, el proceso iniciado no reaccionará más a la señal SIGHUP, y por lo tanto podrá salir del shell, el comando seguirá su ejecución.

Por defecto, los canales de salida y error estándares se redirigen hacia un fichero **nohup.out**, salvo si se precisa la redirección de manera explícita.

```
$ nohup ls -lR / &
10206
$ Sending output to nohup.out
```

► Cuando un proceso hijo finaliza, manda una señal SIGCHLD a su proceso padre. Salvo caso excepcional (un proceso padre desvinculado del hijo), el proceso padre debe obtener tantos SIGCHLD como hijos inició o como SIGHUP emitió. Si el padre termina antes que los hijos, éstos se convertirán en procesos zombis: las señales SIGCHLD son enviadas a... nadie. El proceso hijo ha finalizado correctamente, ha muerto, no consume ningún recurso. No se lo puede matar (ya que está muerto), pero sigue ocupando una entrada en la tabla de los procesos. init lo recupera, y como está siempre en espera, el zombi puede terminar por desaparecer.

## 8. nice y renice

El comando **nice** permite iniciar un comando con una prioridad más baja para permitir a otros posibles procesos ejecutarse más rápidamente.

```
nice [-valor] comando [argumentos]
```

Un valor positivo causará una bajada de prioridad; uno negativo, el aumento de la prioridad (si está autorizado). El valor debe estar comprendido entre -20 y 20. Cuanto más elevado es el valor, menor es la prioridad.

```
$ nice -10 ls -lR / >lista 2>/dev/null&
10884
$ ps -l
  F S      UID      PID    PPID      C PRI  NI   ADDR     SZ WCHAN     TTY
    TIME CMD
80808001 U N+    75  10884  10822 28.5  54  10        0 848K aa3b3a9c tttyp4
  0:03.32 ls
```

comando **renice** funciona un poco como nice, pero permite modificar la prioridad en función de un usuario, un grupo o un PID. El proceso debe estar ya ejecutándose.

```
renice [-n prio] [-p] [-g] [-u] ID
```

La prioridad debe estar entre -20 y 20. El usuario estándar sólo puede utilizar los valores entre 0 y 20 que permiten bajar la prioridad. La opción **-p** precisa un PID, **-g** un GID y **-u** un UID.

## 9. time

El comando **time** mide las duraciones de ejecución de un comando, lo que es ideal para conocer los tiempos de ejecución, y devuelve tres valores:

- **real**: duración total de ejecución del comando;

- **user**: duración del tiempo de CPU necesario para ejecutar el programa;
- **system**: duración del tiempo de CPU necesario para ejecutar los comandos relacionados con el OS (llamadas al sistema dentro de un programa).

El resultado aparece por el canal de error estándar 2. Se puede tener una indicación de la carga de la máquina por el cálculo real / (user+system). Si el resultado es inferior a 10, la máquina dispone de un buen rendimiento; más allá de 20, la carga de la máquina es demasiado pesada (rendimiento reducido).

```
$ time ls -lR /home
...
real    4.8
user    0.2
sys     0.5
```

# Más todavía del bash

## 1. Alias

Un alias es un atajo a un comando y a sus posibles parámetros. Se define con el comando **alias**. Utilizado sin argumentos, lista los alias disponibles.

```
$ alias
alias ..='cd ..'
alias ...='cd ../..'
alias cd..='cd .; '
alias dir='ls -l'
alias l='ls -alF'
alias la='ls -la'
alias ll='ls -l'
alias ls='ls $LS_OPTIONS'
alias ls-l='ls -l'
alias md='mkdir -p'
alias o='less'
alias rd='rmdir'
...
```

Usted puede crear sus propios alias.

```
$ alias deltree='rm -rf'
```

## 2. Agrupación de comandos

El encadenamiento de comandos es posible con «;». También es posible agrupar los comandos. Cuando ejecuta los comandos siguientes:

```
$ uname -a ; pwd ; ls -l >resultado.txt &
```

Se ejecuta únicamente el último comando en segundo plano y sólo se redirecciona su resultado en el fichero resultado.txt. Una solución sería:

```
$ uname -a >resultado.txt & ; pwd >>resultado.txt & ; ls -l >>resultado.txt &
[1] 18232
[2] 18238
[3] 18135
```

Es una solución compleja y no funcionará siempre. Además, incluso si se inician los comandos de manera secuencial, se ejecutan todos en paralelo. El primero que finalice será el primero en escribir en el fichero. La solución consiste en utilizar paréntesis.

```
$ (uname -a ; pwd ; ls -l) > resultado.txt &
[1] 18239
$ [1] Done ( uname -a; pwd; ls -l ) > resultado.txt
```

En todos los casos, se inician todos los comandos colocados entre paréntesis con un subshell, que luego ejecutará los comandos precisados de manera secuencial tal como esté indicado. Así, la redirección concierne al conjunto de los comandos y nada impide lanzar este subshell en segundo plano. De hecho, se distingue claramente un solo PID 18239 durante la ejecución de los comandos.

Otra posibilidad consiste en utilizar llaves {...}. En este caso, no se ejecutará ningún subshell, y si se ejecuta un comando interno (cd u otro), concierne al shell activo. Se debe colocar la llave de cierre justo después de un ;.

```
$ { uname -a; pwd; ls -l; } > resultado.txt
```

Se puede comprobar la diferencia fácilmente entre las dos sintaxis con exit. El primer ejemplo parece no hacer nada, manteniéndose en el shell hijo. El segundo sale de su shell.

```
$ (exit)  
$ { exit; }
```

 Cuidado con los paréntesis, en particular en programación. Como se lanza la agrupación dentro de otro proceso, las posibles variables modificadas dentro de la agrupación no serán visibles una vez terminada la ejecución.

### 3. Relación y ejecución condicional

Además del encadenamiento clásico, se puede relacionar y ejecutar los comandos de manera condicional. La condición de ejecución de un comando es el éxito o no del anterior. Una vez ejecutado, cada comando devuelve un código de retorno, en general 0 si todo ha salido bien, 1 o 2 en caso de error. El shell puede recuperar este valor con la variable \$?.

```
$ ls  
...  
$ echo $?  
0
```

Los caracteres **&&** y **||** permiten efectuar una ejecución condicional.

```
comando1 && comando2  
comando1 || comando2
```

Se ejecutará el comando colocado después de **&&** únicamente si el comando anterior ha devuelto 0 (éxito). Sólo se ejecutará el comando colocado después de **||** si el comando anterior ha devuelto algo diferente a 0.

```
$ grep "ratón" lista && echo "Ratón encontrado" || echo "Ratón no encontrado"  
ratón óptico 30      15  
Ratón encontrado  
$ grep "memoria" lista && echo "Memoria encontrada" || echo "Memoria  
imposible de encontrar"  
Memoria imposible de encontrar
```

# Requisitos y objetivos

## 1. Requisitos

- Saber utilizar el shell y sus comandos.
- Disponer de un editor de texto.
- Estar en posesión de la contraseña root.
- Disponer de uno o varios soportes físicos (discos, llaves) donde crear particiones.
- Conocer los principios del funcionamiento del hardware de un ordenador.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Reconocer los discos y las particiones.
- Manejar los parámetros de los periféricos.
- Elegir entre los diferentes sistemas de ficheros.
- Crear particiones.
- Crear un sistema de ficheros.
- Crear puntos de montaje y modificar el fichero fstab.
- Controlar, modificar y reparar un sistema de ficheros.
- Crear espacios de swap.
- Instalar cuotas.
- Gestionar los permisos y los propietarios de los ficheros.
- Conocer el principio de los hard links.

# Las cuotas de disco

## 1. Definiciones

Las **cuotas** permiten poner límites al uso de espacio en los sistemas de ficheros. Estos límites son de dos tipos:

- **inodos**: limita el número de ficheros.
- **bloques**: limita número de bloques.

Se implementan las cuotas por sistema de ficheros individual, y no para el conjunto de los sistemas de ficheros. Se puede gestionar cada usuario de manera totalmente independiente. Ocurre lo mismo con los grupos. Para cada utilización (inodo o bloque), puede instalar dos límites en el tiempo:

- **Límite duro** (hard): cantidad máxima de inodos o bloques utilizados que el usuario o el grupo no podrá superar en ningún caso. Este límite es taxativo (no podrá crear un fichero cuyo tamaño supera el límite).
- **Límite suave** (soft): cantidad máxima de inodos o bloques utilizados que el usuario o el grupo pueden superar de manera temporal. En este caso, las creaciones y modificaciones serán posibles hasta cierto punto marcado por límite duro o el período de gracia.
- **Período de gracia**. Durante este tiempo el usuario puede continuar trabajando en el sistema de ficheros. El objetivo es que regrese al límite suave en dicho plazo. Una vez superado el período de gracia, el límite suave se convierte en el límite duro. Pase lo que pase, el usuario no podrá nunca superar el límite duro.

Las cuotas se establecen a nivel de núcleo y de sistemas de ficheros. Para utilizarlas, se deben instalar las herramientas de cuotas (paquetes de cuota). En el ejemplo siguiente se muestra cómo instalar las herramientas en un sistema de ficheros ext3.

## 2. Instalación

Va a instalar las cuotas en la partición /home conforme a las etapas siguientes:

- Modifique las opciones de partición en `/etc/fstab`. Añada las opciones `usrquota` (usuario) o `grpquota` (grupo), o las dos.

```
LABEL=/home /home ext3 defaults,usrquota 1 2
```

- Monte de nuevo el sistema de ficheros.

```
# mount -o remount /home
```

- Cree los ficheros que contienen la información de cuota (base de datos de cuotas).

```
# cd /home  
# touch aquota.user aquota.group
```

- Actualice la base de datos con el comando **quotacheck**.

```
# quotacheck -c /home
```

- Inicie (o pare) las cuotas. Esta operación no es necesaria después de un nuevo inicio de Linux, ya que la

implementación de cuotas está incluida en los scripts de inicio. El comando **quotaon** inicia las cuotas para el sistema de ficheros indicado (-a para todos). El comando **quotaoff** deshabilita las cuotas.

```
quotaon /home
```

- Edite las cuotas para los usuarios o los grupos. Se utiliza el comando **edquota**. En la práctica, si va a establecer las mismas cuotas a todos los usuarios, se crea un usuario lambda cuyas propiedades se copiarán para los demás.

Establecer las cuotas para josé:

```
# edquota josé # = edquota -u josé
```

Las cuotas de arturo son idénticas a las de josé:

```
# edquota -p josé arturo
```

- Establezca el período de gracia. El plazo acepta las unidades «seconds», «minutes», «hours», «days», «weeks», «months».

```
# edquota -t
```

- Verifique las cuotas. Los usuarios pueden comprobar el estado de sus cuotas con el comando **quota**. El administrador puede generar un informe con **repquota**. Finalmente, el comando **warnquota**, que puede ser programado con el cron, puede mandar un mail a los usuarios para avisarles en caso de superación.

La edición de las cuotas se hace con el editor por defecto del sistema, que suele ser vi (se puede modificar el comportamiento vía las variables EDITOR y VISUAL). Los bloques de cuotas son bloques de 1KB.

```
# edquota josé
Disk quotas for user josé (uid 502):
Filesystem blocks soft hard inodes soft hard
/dev/hda5 1695256 2500000 3000000 12345 0 0
```

Con el editor se pueden modificar los valores soft y hard que corresponden a los límites suaves y duros para el número de bloques y el número de inodos. Más arriba, se ha establecido un límite suave de unos 2,4 GB (2500000 KB) y duro de unos 2,9 GB (3000000 KB) de ocupación del sistema de ficheros para josé. No hay cuotas en el número de inodos (valor a 0).

 El contenido de los campos blocks e inodos es dinámico y no se debe tocar, ya que de todas maneras no surte ningún efecto.

```
# edquota -t
Filesystem Block grace period Inode grace period
/dev/hda3 7days 7days

# repquota /home
*** Report for user quotas on device /dev/hda5
Block grace time: 7 days; Inode grace time: 7 days
Block limits File limits
User Used soft hard grace used soft hard
-----
root -- 12345 0 0 5 0 0
roger -- 1695256 2500000 3000000 12345 0 0
```

Finalmente, se hace necesario utilizar de forma regular el comando **quotacheck** para mantener la coherencia de la información de cuotas de los sistemas de ficheros. Efectivamente, al deshabilitar las cuotas o detener el

sistema de forma abrupta, resulta necesario verificar y volver a actualizar la información.

```
# quotacheck -avug
```

# Los permisos de acceso

## 1. Los permisos básicos

### a. Permisos y usuarios

El papel de un sistema operativo es también el de asegurar la integridad y el acceso a los datos, lo que es posible gracias a un sistema de permisos. A cada fichero o directorio se le asignan unos privilegios que le son propios, así como autorizaciones de acceso individuales. En el momento intentar el acceso, el sistema comprueba si está autorizado.

Cuando el administrador crea un usuario, le asigna un **UID** (*User Identification*) único. Los usuarios quedan definidos en el fichero `/etc/passwd`. Del mismo modo, cada usuario es integrado en, al menos, un grupo (grupo primario). Todos éstos tienen un identificador único, el **GID** (*Group Identification*) y están definidos en el fichero `/etc/group`.

El comando `id` permite obtener esta información. A nivel interno, el sistema trabaja únicamente con los UID y GID, y no con los propios nombres.

```
$ id  
uid=1000(seb) gid=100(users) grupos=7(lp),16(dialout),33(video),  
100(users)
```

Se asocian un UID y un GID a cada fichero (inode) que define su propietario y su grupo con privilegios. Usted asigna permisos al propietario, al grupo con privilegios y al resto de la gente. Se distinguen tres casos:

- UID del usuario idéntico al UID definido para el fichero. Este usuario es propietario del fichero.
- Los UID son diferentes: el sistema comprueba si el GID del usuario es idéntico al GID del fichero. Si es el caso, el usuario pertenece al grupo con privilegios del fichero.
- En los otros casos (ninguna correspondencia): se trata del resto de la gente (others), ni es el propietario, ni un miembro del grupo con privilegios.

d	rwxr-xr-x	29	seb	users	4096	Mar 15 22:13	Documentos
---	-----------	----	-----	-------	------	--------------	------------

En esta línea de la tabla, el directorio Documentos pertenece al usuario seb y al grupo users, y posee los permisos `rwxr-xr-x`.

### b. Significado

Permiso	Significado
<b>General</b>	
r	Readable (lectura).
w	Writable (escritura).
x	Executable (ejecutable como programa).

### Fichero normal

r	Se puede leer el contenido del fichero, cargarlo en memoria, listar y copiarlo.
w	Se puede modificar el contenido del fichero. Se puede escribir dentro. Modificar el contenido no significa poder eliminar el fichero (ver permisos en directorio).
x	Se puede ejecutar el fichero desde la línea de comandos si se trata de un programa binario (compilado) o de un script (shell, perl...).

### Directorio

r	Se pueden listar (leer) los elementos del directorio (catálogo). Sin esta autorización, ls y los criterios de filtro en el directorio y su contenido no serían posibles. No obstante, puede seguir accediendo a un fichero si conoce su ruta de acceso.
w	Se pueden modificar los elementos del directorio (catálogo), y es posible crear, volver a nombrar y suprimir ficheros en este directorio. <b>Es este permiso el que controla el permiso de eliminación de un fichero.</b>
x	Se puede acceder al catálogo por CD y se puede listar. Sin esta autorización, es imposible acceder al directorio y actuar en su contenido, que pasa a estar cerrado.

Así, para un fichero:

rwX	r-X	r--
Permisos para el propietario de lectura, escritura y ejecución.	Permiso para los miembros del grupo de lectura y ejecución.	Permisos para el resto del mundo de lectura únicamente.

## 2. Modificación de los permisos

Cuando se crea, un fichero o un directorio dispone de permisos por defecto. Utilice el comando **chmod**(*change mode*) para modificar los permisos en un fichero o un directorio. Existen dos métodos para modificar estos derechos: mediante símbolos o mediante un sistema octal de representación de permisos. Sólo el propietario de un fichero puede modificar sus permisos (más el administrador del sistema). El parámetro -R cambia los permisos de manera recursiva.

### a. Mediante símbolos

La sintaxis es la siguiente:

```
chmod modificaciones Fic1 [Fic2...]
```

Si hay que modificar los permisos del propietario, utilice el carácter **u**; para los permisos del grupo con

permisos, el carácter **g**; para el resto, el carácter **o**, y para todos, el carácter **a**.

Para añadir permisos, se utiliza el carácter **+**; para retirarlos, el carácter **-**, y para no tener en cuenta los parámetros anteriores, el carácter **=**.

Finalmente, ponga el permiso cuyos símbolos son: **r**, **w** o **x**.

Puede separar las modificaciones con comas y acumular varios permisos en un mismo comando.

```
$ ls -l  
total 0  
-rw-r--r-- 1 seb users 0 mar 21 22:03 fic1  
-rw-r--r-- 1 seb users 0 mar 21 22:03 fic2  
-rw-r--r-- 1 seb users 0 mar 21 22:03 fic3  
$ chmod g+w fic1  
$ ls -l fic1  
-rw-rw-r-- 1 seb users 0 mar 21 22:03 fic1  
$ chmod u=rwx,g=x,o=rw fic2  
$ ls -l fic2  
-rwx--xr-- 1 seb users 0 mar 21 22:03 fic2  
$ chmod o-r fic3  
$ ls -l fic3  
-rw-r----- 1 seb users 0 mar 21 22:03 fic3
```

Si quiere suprimir todos los derechos, no especifique nada después del signo **=**:

```
$chmod o=fic2  
$ ls -l fic2  
-rwx--x--- 1 seb users 0 mar 21 22:03 fic2
```

## b. Sistema octal

La sintaxis es idéntica a la de los símbolos. A cada permiso le corresponde un valor octal, posicional y acumulable. Para codificar tres permisos rwx, hacen falta tres bits: cada uno tomaría el valor 0 o 1 según la presencia o ausencia del derecho.  $2^3 = 8$ , de ahí la notación octal.

- r vale 4.
- w vale 2.
- x vale 1.

La tabla siguiente le ayuda:

Propietario			Grupo			Resto de la gente		
<b>r</b>	<b>w</b>	<b>x</b>	<b>r</b>	<b>w</b>	<b>x</b>	<b>r</b>	<b>w</b>	<b>x</b>
400	200	100	40	20	10	4	2	1

Para obtener el permiso final, basta sumar los valores. Por ejemplo, si quiere rwxrw-rw-, entonces obtiene  $400+200+100+40+10+4+1=755$ , y para rw-r--r--  $400+200+40+4=644$ .

```
$ chmod 755 fic1  
$ chmod 644 fic2  
$ ls -l fic1 fic2  
-rwxr-xr-x 1 seb users 0 mar 21 22:03 fic1  
-rw-r--r-- 1 seb users 0 mar 21 22:03 fic2
```

La modificación octal de los permisos no es sutil y no permite modificar un solo derecho. Es la totalidad de los permisos lo que se ha modificado de una sola vez.

### 3. Máscara de permisos

#### a. Restringir unos permisos de manera automática

En el momento de la creación de un fichero o de un directorio, se les asigna unos permisos automáticamente. Suele ser rw-r--r-- (644) para un fichero y rwxr-xr-x (755) para un directorio. Una máscara de permisos controla estos valores. Se la puede modificar con el comando **umask**. El comando **umask** coge como parámetro un valor octal del cual cada permiso individual será suprimido de los derechos de acceso máximo del fichero o del directorio.

- Por defecto, se crean todos los ficheros con los permisos 666 (rw-rw-rw-).
- Por defecto, se crean todos los directorios con los permisos 777 (rwxrwxrwx).
- Luego se aplica la máscara.
- La máscara es la misma para el conjunto de los ficheros.
- Una máscara no modifica los permisos de los ficheros existentes, sino solamente los de los ficheros creados a partir de este momento.



Los permisos por defecto (máximo) de los ficheros y de los directorios no son idénticos. Es lógico: como el permiso x permite entrar en un directorio, es normal que éste disponga de él por defecto. Este mismo permiso es inútil por defecto en los ficheros: sólo una pequeña minoría de los ficheros son scripts o binarios.

La máscara por defecto es 022, o sea ----w--w-. Para obtener este valor, inserte **umask** sin parámetro.

```
$ umask  
0022
```

#### b. Cálculo de máscara

##### Para un fichero Para un directorio

Predeterminado	rw-rw-rw- (666)
Retirar	----w--w- (022)
Resta	rw-r--r-- (644)

Predeterminado	rwxrwxrwx (777)
Retirar	----w--w- (022)
Resta	rwxr-xr-x (755)

Observe que aplicar una máscara no es sustraer, sino suprimir permisos de los establecidos por defecto, permiso a permiso. Por ejemplo:

Predeterminado	rw-rw-rw- (666)
Retirar	----wxrwx (037)
Resta	rw-r----- (640)

no 629, lo que es imposible en sistema octal...

## 4. Cambiar de propietario y de grupo

Es posible cambiar el propietario y el grupo de un fichero con la ayuda de los comandos **chown** (*change owner*) y **chgrp** (*change group*). El parámetro **-R** cambia la propiedad de manera recursiva.

```
Al  
chown usuario fic1 [Fic2...]  
chgrp grupo fic1 [Fic2...]
```

especificar el nombre de usuario (o de grupo), el sistema comprueba primero su existencia. Usted puede especificar un UID o un GID. En este caso, el sistema no efectuará comprobación alguna.

Para los dos comandos, no se modifican los permisos anteriores ni la ubicación del fichero. Es posible modificar con un solo comando el propietario y el grupo a la vez.

```
chown usuario[:grupo] fic1 [fic2...]  
chown usuario[.grupo] fic1 [fic2...]
```

Sólo root tiene el permiso de cambiar el propietario de un fichero. Pero un usuario puede cambiar el grupo de un fichero si forma parte del nuevo grupo.

```
$ chgrp video fic1  
$ ls -l fic1  
-rwxr-xr-x 1 seb video 0 mar 21 22:03 fic1
```

## 5. Permisos de acceso especiales

### a. SUID y SGID

Es posible establecer **permisos de acceso especiales** para ficheros ejecutables. Estos permisos de acceso extendidos aplicados a un comando permiten sustituir los permisos otorgados al usuario que lo inició por los permisos del propietario o del grupo de pertenencia del comando.

El ejemplo más sencillo es el programa **passwd**, que permite cambiar la contraseña. Si se ejecutara el comando con los permisos de un usuario clásico, **passwd** no podría abrir y modificar los ficheros /etc/passwd y /etc/shadow:

```
$ ls -l /etc/passwd  
-rw-r--r-- 1 root root 1440 feb 24 10:35 /etc/passwd
```

Puede observar que este fichero pertenece a root, y que sólo root puede escribir en él. Un usuario normal no puede leer su contenido sin interactuar. El comando **passwd** no debería, por lo tanto, poder modificar los ficheros. Vea los permisos del comando **passwd** (/bin/passwd o /usr/bin/passwd):

```
> ls -l /usr/bin/passwd  
-rwsr-xr-x 1 root shadow 78208 sep 21 23:06 /usr/bin/passwd
```

Lleva asociado un nuevo permiso: **s** para los permisos del usuario root. Este nuevo atributo permite la ejecución del comando con permisos de acceso extendidos. Durante el tratamiento, se ejecuta el programa con los permisos del propietario del fichero o de su grupo de pertenencia. En el caso de passwd, se inicia con los permisos de root y no del usuario que lo lanzó.

El permiso **s** sobre el usuario se llama **SUID-Bit** (*Set User ID Bit*), y sobre el grupo, **GUID-Bit** (*Set Group ID Bit*).

El comando **chmod** permite ubicar SUID-Bit y GUID-Bit.

```
chmod u+s comando  
chmod g+s comando
```

Los valores octales son 4000 para SUID-Bit y 2000 para SGID-Bit.

```
chmod 4755 comando  
chmod 2755 comando
```

Sólo el propietario o el administrador puede activar esta propiedad. Posicionar SUID-bit o SGID-Bit tiene sentido únicamente si se han establecido los permisos de ejecución previamente (atributo x en el propietario o el grupo). Si éstos no están presentes; se sustituye la **s** por una **S**.

### b. Real / efectivo

En los datos de identificación del proceso, ha podido observar la presencia de **UID y GID reales y efectivos**. Cuando se inicia un comando con un SUID-Bit o un SGID-Bit posicionado, los permisos se ven modificados. El sistema conserva los UID y GID de origen del usuario que inició el comando (UID y GID reales) transmitidos por el padre, los números UID y GID efectivos son los del propietario o del grupo de pertenencia del programa.

P. ej.: pepito (UID=100, GID=100) envía passwd, que pertenece a root (UID=1, GID=1) con SUID-Bit activado.

```
UID real: 100  
GID real: 100  
UID efectivo: 1  
GID efectivo: 100
```

Si se

posiciona sólo SGID-Bit:

```
UID real: 100  
GID real: 100  
UID efectivo: 100  
GID efectivo: 1
```

Hay que subrayar que no se transmiten los SUID-Bit y SGID-bit a los hijos de un proceso. En este caso, se ejecutarán los hijos con los permisos del usuario que inició el comando básico, los UID reales.

### c. Sticky bit

El **sticky bit** (*bit pegajoso*) permite asignar un criterio protector contra el borrado del contenido de un directorio. Imagine un directorio /tmp donde todos los usuarios tienen el permiso de leer y escribir ficheros.

```
$ ls -ld /tmp  
drwxrwxrwx  6 root      system   16384 Ago 14 13:22 tmp
```

En este directorio todo el mundo puede suprimir ficheros, incluyendo los que no le pertenecen (derecho w presente en todas partes y para todos). Si el usuario pepito crea un fichero, el usuario titi puede suprimirlo incluso aunque no le pertenezca.

El sticky bit aplicado a un directorio, aquí /tmp, impide esta operación. Sí, pepito aún puede visualizar y modificar el fichero, pero sólo su propietario (o el administrador) podrá suprimirlo.

```
$ chmod u+t /tmp  
ls -ld /tmp
```

```
| drwxrwxrwt 35 root root 77824 mar 21 22:30 /tmp
```

En octal, se utilizará el valor 1000 (`chmod 1777 /tmp`).

Aunque aplicado al usuario, el sticky bit, representado por una **t**, aparece en el grupo de permisos de «others».

#### d. Permisos y directorio

Si usted da el permiso s al grupo en un directorio, todos los ficheros creados dentro de este directorio serán del mismo grupo que este directorio, sea cual sea el grupo de la persona que crea este fichero.

```
$ mkdir dir
$ chmod 770 dir
$ ls -ld dir
drwxrwx--- 2 seb users 4096 mar 21 22:36 dir
$ chgrp video dir
$ chmod g+s dir
$ ls -ld dir
drwxrws--- 2 seb video 4096 mar 21 22:37 dir
$ cd dir
$ touch pepito
$ ls -l pepito
-rw-r--r-- 1 seb video 0 mar 21 22:37 pepito
```

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

## **1. Preguntas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **Discos y sistemas de ficheros**

**1** ¿Qué representa el periférico /dev/hde2?

- A - Imposible, sólo pueden existir 4 discos IDE en un PC.
- B - El segundo disco del quinto puerto IDE.
- C - El disco esclavo conectado en el tercer puerto IDE.
- D - La segunda partición del disco maestro del controlador IDE2.

**2** ¿Cómo se ve un disco conectado en SATA en una máquina cuyo SATA nativo es soportado por Linux?

**3** ¿Permite el comando **lsscsi** diferenciar el «verdadero» SCSI de los discos SATA o USB?

**4** ¿Cómo obtener la información detallada relativa el disco /dev/hda?

- A - cat /dev/hda
- B - hdparm -I /dev/hda
- C - getparm /dev/hda
- D - sddparm /dev/hda

**5** ¿Dónde se almacenan los metadatos de un fichero?

**6** ¿Por qué un sistema de ficheros con diario es preferible a los otros?

- A - Porque es más reciente.
- B - Porque es más rápido.
- C - Porque es más fiable.
- D - Porque no hay nada más.

**7** ¿Es posible utilizar un sistema de ficheros FAT o VFAT como sistema de partición principal (raíz) de un sistema Linux?

**8** ¿Es pertinente utilizar xfs en un puesto de trabajo?

**9** ¿Dónde se sitúa el MBR?

- A - Al principio del disco.
- B - Al principio de la primera partición primaria.
- C - Al principio de la partición extendida.
- D - Al principio de la primera partición lógica.

### **Particiones**

**10** ¿Cómo crear más de cuatro particiones en su disco?

- A - Imposible: sólo puede haber 4 particiones.
- B - Creando una partición extendida que contenga otras particiones lógicas.

- C - Añadiendo un disco y creando un LVM.
- D - Al suprimir la swap se puede añadir una partición.

**11** ¿Cuál es el tipo hexadecimal de partición que se debe crear para colocar en ella un sistema de ficheros ext3?

- A - fd
- B - 8e
- C - 88
- D - 82

**12** ¿Qué parámetro se debe pasar a fdisk para listar las particiones del disco /dev/sdb?

**13** En un disco /dev/hdb vacío, ¿cuál es la secuencia completa bajo fdisk para crear una partición primaria de tipo Linux que utiliza todo el espacio?

**14** Linux aún no ve la partición que usted ha creado. ¿Cómo actualizar la tabla de las particiones del núcleo?

- A - partprobe /dev/hdb
- B - cat /proc/partitions
- C - touch /dev/hdb1
- D - Hay que reiniciar obligatoriamente.

### **Creación del sistema de ficheros**

**15** Si un sistema de ficheros tiene un tamaño de bloque de 4096 bytes, ¿qué lugar ocupa un fichero de 1024 bytes?

- A - Un cuarto de bloque.
- B - 1 KB.
- C - Un bloque.
- D - El sistema completará el fichero para ocupar 4096 bytes.

**16** ¿Cuántos superbloques existen en un sistema de ficheros ext3?

- A - Al menos 2.
- B - Al menos 4.
- C - Imposible saberlo.
- D - Depende del tamaño del sistema de ficheros.

**17** ¿Cuál es la única información de un fichero que falta en el inodo?

**18** Usted suprime un fichero cuyo contador de vínculos estaba a 2. ¿Ha desaparecido totalmente?

**19** ¿Es fácil recuperar un fichero suprimido?

**20** ¿Qué línea de comandos utilizar para crear una partición ext3 dotada del label «PUBLIC» en la partición /dev/hdb1?

### **Montaje**

**21** ¿Qué comando debe insertar para acceder a la partición /dev/hdb1 desde /media/PUBLIC?

**22** ¿Qué línea debe añadir en /etc/fstab para que la partición anterior sea montada automáticamente en /media/PUBLIC en arranque, teniendo como referencia su nombre de volumen y con los ACL activados?

**23** Como root, intente desmontar el sistema de ficheros raíz. ¿Por qué obtiene este resultado?

## **Gestión del sistema de ficheros**

**24** ¿Cómo conocer el tamaño en MB utilizado en el sistema de ficheros montado en /home?

- A - du -s /home
- B - df -kP /home
- C - free /home
- D - df -m /home

**25** El sistema de ficheros raíz contiene a menudo otros puntos de montaje, como /home o /opt, /usr, etc., según su instalación. ¿Cómo conocer la ocupación de la totalidad de los ficheros y directorios desde la raíz sin los otros puntos de montaje?

**26** Durante el inicio del sistema, no se pudo montar un sistema de ficheros contenidos en /de/sda3. ¿Qué comando debe ejecutar para comprobar y arreglar su contenido?

**27** Está a punto de volver a poner en servicio un disco que sufrió un accidente o que estuvo guardado durante mucho tiempo. Linux lo reconoce como /dev/sdd. ¿Qué comando puede insertar para comprobar el estado del disco?

- A - fsck -c /dev/sdd
- B - mkfs -t ext3 -c /dev/sdd
- C - badblocks -vw /dev/sdd
- D - badblocks -v /dev/sdd

**28** ¿Cómo conocer el tamaño en bytes de un bloque del sistema de ficheros ext3 en /dev/hdb1?

**29** Conecta un antiguo disco duro en IDE1, esclavo, en el cual la primera partición primaria está en ext2. ¿Cómo convertirla en ext3?

**30** Su servidor dispone de un disco SCSI (ID: 2) que contiene una partición de 600 GB, con un sistema de ficheros ext3. Este servidor se reinicia pocas veces. Lleva más de 400 días funcionando. Ahora bien, usted debe reiniciar en emergencia este servidor tras una actualización del núcleo. Debe volver a ponerlo en producción lo antes posible, pero se iniciará el comando **fsck** y idurará horas! ¿Cómo evitar el check en el momento del próximo boot?

- A - Comenta la línea correspondiente en /etc/fstab y vuelve a subir manualmente después delboot.
- B - Desactiva la verificación vía tune2fs -i 0 -c 0 /dev/sdb1.
- C - Efectúa un check vía fsck -f /dev/sdb1 antes de reiniciar.
- D - Es una mala idea: el sistema de ficheros lleva demasiado tiempo sin control.

## **La swap y la memoria**

**31** Su puesto de trabajo dispone de 512 MB de memoria. ¿Qué tamaño de swap va a crear?

**32** ¿Cómo crear una swap en la partición /dev/hdb3?

**33** Se ha equivocado de tamaño al crear la partición de swap. ¿Cómo desactivarla?

- A - swapstop /dev/hdb3
- B - rmswap /dev/hdb3
- C - swapoff /dev/hdb3
- D - No se puede: hay que suprimir la línea de fstab y reiniciar.

**34** ¿Cómo obtener la lista de todas las particiones y ficheros de swap?

**35** ¿Cómo visualizar el estado de la memoria en MB?

**36** Aunque usted dispone de 1 GB de memoria, Linux le informa de que sólo quedan 30 MB libres. Sin

embargo, no observa ninguna utilización de la swap, incluso cuando ejecuta otros programas. ¿Qué está pasando?

**37** ¿Qué posibilidades tiene si resulta que le falta memoria?

- A - Cambiar su ordenador por un modelo más reciente.
- B - Ver si es posible aumentar la cantidad de memoria del ordenador.
- C - Añadir una partición o un fichero de swap.
- D - Parar los servicios y programas poco utilizados.

### **Las cuotas de disco**

**38** ¿Qué ocurre si supera el límite suave, pero aún no el límite duro, una vez que ha vencido el período de gracia?

- A - No puede superar el límite alcanzado.
- B - No alcanzó el límite duro, puede proseguir.
- C - Tiene que volver por debajo del límite suave.
- D - Tiene que pedir a su administrador que aumente sus límites.

**39** ¿Qué se debe añadir a una línea de /etc/fstab y dónde, para activar las cuotas?

**40** ¿Tiene que activar las cuotas después de cada reinicio?

**41** ¿Cómo editar las cuotas del usuario seb?

**42** ¿Cuáles son los dos tipos de cuotas que puede modificar?

- A - El número de ficheros.
- B - El número de directorios.
- C - El tamaño en kilobytes.
- D - El tamaño en bytes.

**43** Tuvo que desactivar de manera temporal las cuotas para efectuar el mantenimiento de los sistemas de ficheros. ¿Cómo activarlas de nuevo?

### **Los permisos**

**44** En un fichero, ¿indica el permiso x que un fichero es obligatoriamente un programa?

**45** Dispone del permiso w de escritura en un fichero. ¿Esto le da el permiso de suprimirlo?

**46** Un directorio que le pertenece está en otro directorio en el cual el permiso x está ausente. ¿Puede acceder a su directorio?

- A - Sí: me pertenece, por lo tanto puedo acceder a él.
- B - Sí: ya que me pertenece, el directorio que lo contiene también.
- C - No: el directorio que lo contiene no me pertenece.
- D - No, no tengo permiso de cruzar el directorio que lo contiene.

**47** Usted es el propietario del fichero fic, que ha retirado a los miembros del grupo y otros todos los permisos sobre él. Sin embargo, el directorio que lo contiene dispone del permiso w para los miembros del grupo. ¿Quién puede suprimir su fichero?

- A - Usted solamente.
- B - Usted y los miembros del grupo del directorio.
- C - Nadie: suprimió todos los permisos del fichero.
- D - Los demás.

**48** Con chmod, ¿cuáles son los parámetros que se deben colocar para suprimir los permisos de escritura de los demás, dar el permiso de lectura y ejecución a los miembros del grupo y todos los permisos a usted?

**49** Convierta los permisos rwxr-x-- en octal.

**50** ¿Cuál sería la máscara más severa para todo el mundo excepto usted?

- A - 700
- B - 066
- C - 077
- D - 022

**51** ¿Cuál es el efecto de la máscara 011 en un fichero? Indique los permisos del fichero en símbolos.

**52** Usted forma parte de los grupos «users» y «audio». ¿Tiene permiso de cambiar el grupo de uno de sus fichero a «vídeo»?

**53** Un comando de conexión a un módem necesita poder leer y escribir en un periférico que pertenece a root. El comando pertenece también a root. Usted puede:

- A - Dar los permisos r y w del periférico a todo el mundo.
- B - Dar el permiso w al grupo del periférico y el permiso SGID al programa del mismo grupo.
- C - Dar el permiso SUID al programa del mismo propietario que el periférico.
- D - Hacer que su usuario pertenezca al mismo grupo que el periférico.

**54** Quiere crear un directorio donde todo el mundo pueda escribir ficheros, donde todos los ficheros tengan el mismo grupo, pero donde nadie pueda suprimir los ficheros de los demás. ¿Cuáles deben ser los derechos?

## 2. Resultados

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos: /54

Para este capítulo, su resultado mínimo debe ser de 40 respuestas acertadas (40/54).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- Discos y sistemas de ficheros.
- Particiones.
- Creación del sistema de ficheros.
- Montaje.
- Gestión del sistema de ficheros.
- La swap y la memoria.
- Las cuotas de disco.
- Los permisos.

## 3. Respuestas

**Discos y sistemas de ficheros**

**1** ¿Qué representa el periférico /dev/hde2?

- A - Imposible, sólo pueden existir 4 discos IDE en un PC.
- B - El segundo disco del quinto puerto IDE.
- C - El disco esclavo conectado en el tercer enchufe IDE.
- D - La segunda partición del disco maestro del controlador IDE2.

*D. Puede añadir tantas tarjetas IDE como desee.*

**2** ¿Cómo se ve un disco conectado en SATA en una máquina cuyo SATA nativo es soportado por Linux?

*Como un disco SCSI.*

**3** ¿Permite el comando **lsscsi** diferenciar el «verdadero» SCSI de los discos SATA o USB?

*Sí. La tercera columna puede indicar USB si el fabricante es desconocido. Lo mejor es utilizar el parámetro -v, que indicará la ruta física al periférico (/sys/devices/.../usbX/...).*

**4** ¿Cómo obtener la información detallada relativa el disco /dev/hda?

- A - cat /dev/hda
- B - hdparm -I /dev/hda
- C - getparm /dev/hda
- D - sdparm /dev/hda

*B. Utilice hdparm para un disco IDE o SATA. Utilice sdparm para un disco SCSI.*

**5** ¿Dónde se almacenan los metadatos de un fichero?

*En su inodo.*

**6** ¿Por qué un sistema de ficheros con diario preferible a los otros?

- A - Porque es más reciente.
- B - Porque es más rápido.
- C - Porque es más fiable.
- D - Porque no hay nada más.

*C. El diario asegura la mejor integridad posible de los datos en caso de avería.*

**7** ¿Es posible utilizar un sistema de ficheros FAT o VFAT como sistema de partición principal (raíz) de un sistema Linux?

*No. FAT no gestiona los permisos y la seguridad, lo que hace imposible su utilización como sistema de ficheros principal. Utilice FAT para la compatibilidad con Windows.*

**8** ¿Es pertinente utilizar xfs en un puesto de trabajo?

*No. Las numerosas posibilidades que ofrece xfs están aún hoy sin explotar.*

**9** ¿Dónde se sitúa el MBR?

- A - Al principio del disco.
- B - Al principio de la primera partición primaria.

- C - Al principio de la partición extendida.
- D - Al principio de la primera partición lógica.

A. Son los 512 primeros bytes del disco.

### **Particiones**

**10** ¿Cómo crear más de cuatro particiones en su disco?

- A - Imposible: sólo puede haber 4 particiones.
- B - Creando una partición extendida que contenga otras particiones lógicas.
- C - Añadiendo un disco y creando un LVM.
- D - Al suprimir la swap, se puede añadir una partición.

B. Hasta 15 o 63 particiones son posibles.

**11** ¿Cuál es el tipo hexadecimal de partición que se debe crear para colocar en ella un sistema de ficheros ext3?

- A - fd
- B - 8e
- C - 88
- D - 82

D. Cree una partición con el tipo genérico Linux, o sea, 82 en hexadecimal.

**12** ¿Qué parámetro se debe pasar a fdisk para listar las particiones del disco /dev/sdb?

El -l: `fdisk -l /dev/sdb`

**13** En un disco /dev/hdb vacío, ¿cuál es la secuencia completa bajo fdisk para crear una partición primaria de tipo Linux que utiliza todo el espacio?

`fdisk /dev/hdb. Teclee: n (new), p (primaria), 1 (primera partición primaria), Entrada (1er cilindro), Entrada (último cilindro) luego w (guardar).`

**14** Linux aún no ve la partición que usted ha creado. ¿Cómo actualizar la tabla de las particiones del núcleo?

- A - partprobe /dev/hdb
- B - cat /proc/partitions
- C - touch /dev/hdb1
- D - Hay que reiniciar obligatoriamente.

A. Si partprobe no está presente, instale parted o en el caso contrario vuelva a bootear.

### **Creación del sistema de ficheros**

**15** Si un sistema de ficheros tiene un tamaño de bloque de 4096 bytes, ¿qué lugar ocupa un fichero de 1024 bytes?

- A - Un cuarto de bloque.
- B - 1 KB.

- C - Un bloque.
- D - El sistema completará el fichero para ocupar 4096 bytes.

C. *El espacio sobrante del bloque se desaprovecha.*

**16** ¿Cuántos superbloques existen en un sistema de ficheros ext3?

- A - Al menos 2.
- B - Al menos 4.
- C - Imposible saberlo.
- D - Depende del tamaño del sistema de ficheros.

A y D. *Siempre hay al menos una copia del superbloque, pero sólo un comando dumpe2fs le dará el número exacto.*

**17** ¿Cuál es la única información de un fichero que falta en el inodo?

*Su nombre. Está en la tabla catálogo.*

**18** Usted suprime un fichero cuyo contador de vínculos estaba a 2. ¿Ha desaparecido totalmente?

*No, el contador está a 1, un nombre se coloca en otra parte en otra tabla catálogo.*

**19** ¿Es fácil recuperar un fichero suprimido?

*No. Cuando se destruye el inodo, se borran los campos correspondientes que apuntan a los bloques de datos. Entonces habría que analizar todo el disco bloque por bloque para encontrar las partes. Hay herramientas especializadas (incluso en Linux), pero no están al alcance del usuario corriente.*

**20** ¿Qué línea de comandos utilizar para crear una partición ext3 dotada del label «PUBLIC» en la partición /dev/hdb1?

`mkfs -t ext3 -L PUBLIC /dev/hdb1`

### **Montaje**

**21** ¿Qué comando debe insertar para acceder a la partición /dev/hdb1 desde /media/PUBLIC?

`mount -t ext3 /dev/hdb1 /media/PUBLIC`

**22** ¿Qué línea debe añadir en /etc/fstab para que la partición anterior sea montada automáticamente en /media/PUBLIC en arranque, teniendo como referencia su nombre de volumen y con los ACL activados?

`LABEL=PUBLIC/media/PUBLIC ext3 acl1 2`

**23** Como root, intente desmontar el sistema de ficheros raíz. ¿Por qué obtiene este resultado?

`umount /. No funciona, ya que el sistema de ficheros está siendo utilizado.`

### **Gestión del sistema de ficheros**

**24** ¿Cómo conocer el tamaño en MB utilizado en el sistema de ficheros montado en /home?

- A - du -s /home
- B - df -kP /home
- C - free /home

- D - df -m /home

*D. Es df, y no du, ya que se trata del sistema de ficheros, y no del conjunto de la estructura donde se pueden montar otros sistemas de ficheros.*

**25** El sistema de ficheros raíz contiene a menudo otros puntos de montaje, como /home o /opt, /usr, etc., según su instalación. ¿Cómo conocer la ocupación de la totalidad de los ficheros y directorios desde la raíz sin los otros puntos de montaje?

*du -x /. El resultado necesitará decenas de páginas, ya que se detalla fichero por fichero, salvo para los otros sistemas de ficheros.*

**26** Durante el inicio del sistema, no se pudo montar un sistema de ficheros contenidos en /de/sda3. ¿Qué comando debe ejecutar para comprobar y arreglar su contenido?

*fsck -V /dev/sda3*

**27** Está a punto de volver a poner en servicio un disco que sufrió un accidente o que estuvo guardado durante mucho tiempo. Linux lo reconoce como /dev/sdd. ¿Qué comando puede insertar para comprobar el estado del disco?

- A - fsck -c /dev/sdd
- B - mkfs -t ext3 -c /dev/sdd
- C - badblocks -vw /dev/sdd
- D - badblocks -v /dev/sdd

*D, aunque es preferible C, ya que se efectúa una prueba de escritura. Cuidado, requiere mucho tiempo.*

**28** ¿Cómo conocer el tamaño en bytes de un bloque del sistema de ficheros ext3 en /dev/hdb1?

*dump2fs -h /dev/hdb1 | grep -i "block size"*

**29** Conecta un antiguo disco duro en IDE1, esclavo, en el cual la primera partición primaria está en ext2. ¿Cómo convertirla en ext3?

*Con tune2fs: tune2fs -j /dev/hdd1*

**30** Su servidor dispone de un disco SCSI (ID: 2) que contiene una partición de 600 GB, con un sistema de ficheros ext3. Este servidor se reinicia pocas veces. Lleva más de 400 días funcionando. Ahora bien, usted debe reiniciar en emergencia este servidor tras una actualización del núcleo. Debe volver a ponerlo en producción lo antes posible, pero se iniciará el comando **fsck** y tardará horas! ¿Cómo evitar el check en el momento del próximo boot?

- A - Comenta la línea correspondiente en /etc/fstab y vuelve a subir manualmente después del boot.
- B - Desactiva la verificación vía **tune2fs -i 0 -c 0 /dev/sdb1**.
- C - Efectúa un check vía **fsck -f /dev/sdb1** antes de reiniciar.
- D - Es una mala idea: el sistema de ficheros lleva demasiado tiempo sin control.

*Lo lógico sería contestar D. Sin embargo, le corresponde a usted decidir: programar la parada en un hueco cuando el sistema no está saturado o desactivar de manera temporal (y volver a ponerlo después) el control: respuesta B. Inútil comentar la línea. En el montaje se haría la comprobación. No fuerce la verificación en un sistema de ficheros montado.*

## **La swap y la memoria**

**31** Su puesto de trabajo dispone de 512 MB de memoria. ¿Qué tamaño de swap va a crear?

*Al menos 1 GB.*

**32** ¿Cómo crear una swap en la partición /dev/hdb3?

```
mkswap /dev/hdb3
```

**33** Se ha equivocado de tamaño al crear la partición de swap. ¿Cómo desactivarla?

- A - swapstop /dev/hdb3
- B - rmswap /dev/hdb3
- C - swapoff /dev/hdb3
- D - No se puede: hay que suprimir la línea de fstab y reiniciar.

*C, si la swap no está en uso. De lo contrario, reiniciar el sistema en modo Single.*

**34** ¿Cómo obtener la lista de todas las particiones y ficheros de swap?

```
cat /proc/swaps
```

**35** ¿Cómo visualizar el estado de la memoria en MB?

```
free -m
```

**36** Aunque dispone de 1 GB de memoria, Linux le informa de que sólo quedan 30 MB libres. Sin embargo, no observa ninguna utilización de la swap, incluso cuando ejecuta otros programas. ¿Qué está pasando?

*Mire el estado de la caché y de los buffers. Uno de ellos utiliza probablemente una cantidad importante de memoria. El núcleo liberará esta memoria en caso de necesidad. Por defecto, el núcleo intenta utilizar la mayor memoria posible.*

**37** ¿Qué posibilidades tiene si resulta que le falta memoria?

- A - Cambiar su ordenador por un modelo más reciente.
- B - Ver si es posible aumentar la cantidad de memoria del ordenador.
- C - Añadir una partición o un fichero de swap.
- D - Parar los servicios y programas poco utilizados.

*Pueden ser todas válidas, pero si le falta realmente memoria, sería más ventajoso gastarse unas decenas de euros en comprar más memoria para el ordenador si dispone de slots de expansión libres.*

### **Las cuotas de disco**

**38** ¿Qué ocurre si supera el límite suave, pero aún no el límite duro, una vez que ha vencido el período de gracia?

- A - No puede superar el límite alcanzado.
- B - No alcanzó el límite duro, puede proseguir.
- C - Tiene que volver por debajo del límite suave.
- D - Tiene que pedir a su administrador que aumente sus límites.

*A y C. Ya no puede superar el límite que ha alcanzado. Tiene que regresar por debajo del límite suave para tener de nuevo un período de gracia.*

**39** ¿Qué se debe añadir a una línea de /etc/fstab y dónde, para activar las cuotas?

Tiene que añadir las opciones `usrquota` o `grpquota`, o las dos, según el tipo de cuota deseado.

**40** ¿Tiene que activar las cuotas después de cada reinicio?

No. Los scripts de inicio lo hacen en su lugar.

**41** ¿Cómo editar las cuotas del usuario seb?

Con el comando `edquota seb`.

**42** ¿Cuáles son los dos tipos de cuotas que puede modificar?

- A - El número de ficheros.
- B - El número de directorios.
- C - El tamaño en kilobytes.
- D - El tamaño en bytes.

A y D. Un directorio es un fichero. Utiliza un inodo. Siempre se especifica el tamaño en bytes.

**43** Tuvo que desactivar de manera temporal las cuotas para efectuar el mantenimiento de los sistemas de ficheros. ¿Cómo activarlas de nuevo?

El comando `quotacheck` permite comprobar y poner en condiciones la base de las cuotas. Las opciones `-a(all)`, `-u(user)`, `-g(group)` y `-v(verbose)` permiten controlarlo todo y ver el resultado.

### **Los permisos**

**44** En un fichero, ¿indica el permiso x que un fichero es obligatoriamente un programa?

No. Es posible poner el permiso x en cualquier fichero, incluso si no es un programa.

**45** Dispone del permiso w de escritura en un fichero. ¿Esto le da el permiso de suprimirlo?

No. Son los permisos del directorio que contiene el fichero que indican si un usuario tiene el permiso suprimir un fichero.

**46** Un directorio que le pertenece está en otro directorio en el cual el permiso x está ausente. ¿Puede acceder a su directorio?

- A - Sí: me pertenece, por lo tanto puedo acceder a ello.
- B - Sí: ya que me pertenece, el directorio que lo contiene también.
- C - No: el directorio que lo contiene no me pertenece.
- D - No, no tengo permiso de cruzar el directorio que lo contiene.

D. El permiso x que falta impide cualquier acceso al directorio, incluso si contiene ficheros y directorios que le pertenecen.

**47** Usted es el propietario del fichero `fic`, que ha retirado a los miembros del grupo y otros todos los permisos sobre él. Sin embargo, el directorio que lo contiene dispone del permiso w para los miembros del grupo. ¿Quién puede suprimir su fichero?

- A - Usted solamente.
- B - Usted y los miembros del grupo del directorio.
- C - Nadie: suprimió todos los permisos del fichero.

- D - Los demás.

B. Es el permiso de escritura en el directorio el que autoriza a las personas a modificar su contenido.

48 Con chmod, ¿cuáles son los parámetros que se deben colocar para suprimir los permisos de escritura a los demás, dar el permiso de lectura y ejecución a los miembros del grupo y todos los permisos a usted?

$u=rwx, g=rw, o-w$ .

49 Convierta los permisos rwxr-x-- en octal.

750.

50 ¿Cuál sería la máscara más severa para todo el mundo excepto usted?

- A - 700
- B - 066
- C - 077
- D - 022

C. Retira todos los permisos a los miembros del grupo y a los demás.

51 ¿Cuál es el efecto de la máscara 011 en un fichero? Indique los permisos del fichero en símbolos.

$rw-rw-rw$ . Como el permiso x (1) está ausente por defecto, eso no cambia nada.

52 Usted forma parte de los grupos «users» y «audio». ¿Tiene permiso de cambiar el grupo de uno de sus ficheros a «vídeo»?

No, no forma parte del grupo «vídeo».

53 Un comando de conexión a un módem necesita poder leer y escribir en un periférico que pertenece a root. El comando pertenece también a root. Usted puede:

- A - Dar los permisos r y w del periférico a todo el mundo.
- B - Dar el permiso w al grupo del periférico y el permiso SGID al programa del mismo grupo.
- C - Dar el permiso SUID al programa del mismo propietario que el periférico.
- D - Hacer que su usuario pertenezca al mismo grupo que el periférico.

Prefiera la respuesta D con los permisos adaptados. Evite si es posible los permisos SUID/SGID, inútiles o en programas poco seguros.

54 Quiere crear un directorio donde todo el mundo pueda escribir ficheros, donde todos los ficheros tengan el mismo grupo, pero donde nadie pueda suprimir los ficheros de los demás. ¿Cuáles deben ser los derechos?

$rxrwsrwt$ , o sea, 3777 en octal.

# Prácticas

## 1. Los discos y particiones

Objetivo: crear una partición y que el sistema la reconozca. ¡Cuidado: la operación puede ser destructiva!

1. Supongamos un disco duro de su sistema que dispone de espacio para la creación de una partición. En la máquina de prueba, se trata del primer disco IDE, llamado /dev/hda. Hay tres particiones: /dev/hda1, /dev/hda2 y /dev/hda3. Cree una tercera partición extendida.

Inicie fdisk:

```
# fdisk /dev/hda
```

2. Cree una nueva partición pulsando n y luego [Intro].  
Cree una partición extendida pulsando e y luego [Intro].  
Como número, teclee el 4 y luego [Intro]: sólo puede haber cuatro particiones primarias.  
Finalmente, pulse dos veces [Intro] para aceptar los valores por defecto.
3. Cree luego una partición lógica, según el mismo principio, en la partición extendida. Llevará el número 5 y ocupará todo el espacio disponible.
4. Guarde la tabla de las particiones pulsando w y luego [Intro]. Seguramente obtendrá un mensaje de aviso que le informará de que la nueva tabla no está accesible.
5. Fuerce la actualización de la nueva tabla de las particiones con partprobe:

```
# partprobe /dev/hda
```

6. Verifique en /dev la presencia del fichero /dev/hda5.

```
# ls -l /dev/hda5
```

## 2. Creación de un sistema de ficheros

Objetivo: crear y manejar el sistema de ficheros en /dev/hda5. Cuidado: esta operación es destructiva.

1. Cree un sistema de ficheros de tipo ext2 en /dev/hda5, sin opciones particulares:

```
# mkfs -t ext2 /dev/hda5
```

2. La verdad sea dicha, había que crearlo en ext3 para aprovechar su transaccionalidad.  
Modifique el sistema de ficheros para que el tipo sea ext3:

```
# tune2fs -j /dev/hda5
```

3. La nueva partición va a servir al almacenamiento de varios datos. Asígnele una etiqueta (nombre, label): DATOS.

```
# e2label /dev/hda5 DATOS
```

4. Ahora intente conocer el identificador único de sistema de ficheros, el UUID, de su nuevo sistema de ficheros. Dispone de dos métodos:  
Por udev:

```
# /lib/udev/vol_id --uuid /dev/hda5  
527585d3-1e52-4aba-b7fc-70f18388458d
```

Por dumpe2fs:

```
# dumpe2fs /dev/hda6 | grep -i uuid  
Filesystem UUID: 527585d3-1e52-4aba-b7fc-70f18388458d
```

Por blkid:

```
# blkid -o value -s UUID /dev/hda6  
527585d3-1e52-4aba-b7fc-70f18388458d
```

### 3. Acceso y montaje del sistema de ficheros

Objetivo: acceder al nuevo sistema de ficheros creado en la práctica anterior.

1. Cree un directorio que servirá de punto de montaje al nuevo sistema de ficheros:

```
# mkdir /DATOS
```

2. Monte el sistema de ficheros, por su nombre de periférico, en este directorio:

```
# mount -t ext3 /dev/hda5 /DATOS
```

3. Muévase a este punto de montaje y cree un fichero cualquiera:

```
# cd /DATOS  
# touch pepito
```

4. Salga de este directorio (CD) y desmonte el sistema de ficheros:

```
# umount /DATOS
```

5. Añada una línea en /etc/fstab para poder montar este sistema de ficheros de manera automática, por su label.

La línea es:

```
LABEL=DATOS /DATOS ext3 defaults 0 0
```

6. Monte el sistema de ficheros de manera sencilla desde el nombre de su punto de montaje:

```
# mount /DATOS
```

### 4. Estadísticas y mantenimiento del sistema de ficheros

Objetivo: obtener información sobre la ocupación del sistema de ficheros y repararlo si fuese necesario.

1. Mire el estado de ocupación de sus sistemas de ficheros de manera legible para un simple «humano»:

```
# df -H
```

2. El sistema de ficheros que apunta a /home parece ocupado. Se trata de determinar lo que puede ocupar tanto espacio. Determine la ocupación de cada fichero y directorio:

```
# du -m /home
```

3. El resultado es demasiado largo. Ordene la salida de manera que aparezcan las mayores ocupaciones en último lugar:

```
# du -m | sort -n
```

4. Un directorio montado en /DATOS (práctica anterior) tiene problemas: parece que el contenido de un directorio se ha corrompido; nombres de ficheros y tamaños extraños. Verifique y arregle este sistema de ficheros:

```
# cd  
# umount /DATOS  
# fsck /dev/hda5
```

5. Fuerce una verificación de este sistema de ficheros en el próximo reinicio. El manual de tune2fs indica que se trata de la opción -C:

```
# tune2fs -C 1000 /dev/hda5
```

## 5. Swap y memoria

Objetivo: gestionar la swap y la memoria.

1. En una máquina dada, el balance memoria se presenta así, ¿qué deduce de ello?

```
# free  
total used free shared buffers cached  
Mem: 2060680 2011224 49456 0 170628 958508  
-/+ buffers/cache: 882088 1178592  
Swap: 2104472 1296 2103176
```

La máquina dispone de 2 GB de memoria física y de 2 GB de swap. Aunque indica unos 48 MB de memoria libre, hay unos 950 MB de memoria caché y 160 MB de memoria intermedia. O sea, 1 GB de memoria potencial para liberar.

2. Verifique el nombre de la partición que contiene el espacio o los espacios de swap. Un man de swapon indica que se puede encontrar la información en /proc/swaps:

```
# cat /proc/swaps  
Filename Type Size Used Priority  
/dev/sda5 partition 2104472 1336 -1
```

3. A este nivel de carga, posiblemente la swap de esta máquina sea inútil. Desactívuela.

```
# swapoff /dev/sda5
```

4. Unos instantes más tarde, debe cargar una aplicación pesada de tratamiento de imagen que va a consumir mucha memoria. Vuelva a cargar la totalidad de las zonas de swap:

```
# swapon -a
```

5. Consulte la información detallada en la memoria:

```
# cat /proc/meminfo
```

## 6. Cuotas

Objetivo: instalar cuotas.

1. Modifique con vi la línea añadida en /etc/fstab durante la práctica para activar las cuotas de usuario y vuelva a montar el sistema de ficheros:  
En /etc/fstab:

```
LABEL=DATOS /DATOS ext3 defaults,usrquota 0 0
```

Luego:

```
# mount -o remount /DATOS
```

2. Cree y active las cuotas.

```
# quotacheck -c /DATOS  
# quotaon /DATOS
```

3. Coloque un límite global de 150 MB al usuario seb:

```
# edquota seb
```

Luego inscriba el valor 153600 (en bytes) en hard y soft, y guarde.

## 7. Los permisos

Objetivo: modificar los permisos y comprender el mecanismo de los permisos y límites de los usuarios.

1. Modifique los propietarios y grupos de /DATOS en seb:users.

```
# chown seb:users /DATOS
```

2. Todo el mundo debe poder escribir en esta carpeta, pero sin suprimir los ficheros de los demás. Asimismo, todos los ficheros creados en este directorio deben pertenecer al grupo users. Coloque los permisos correctos: hacen falta todos los permisos para todo el mundo, el permiso sticky y el SGID-bit:

```
# chmod 3777 /DATOS
```

3. Cree un directorio de prueba en /tmp con los permisos rwxrwxrwx. Cree un fichero en él y modifique sus permisos retirando el permiso w al grupo y los demás. ¿Quién puede suprimirlo?

```
# mkdir /tmp/test  
# chmod 777 /tmp/test  
# touch /tmp/test/pepito  
# chmod g-w,o-w /tmp/test/pepito
```

Todo el mundo puede suprimir este fichero: sus permisos no tienen importancia. Son los

permisos del directorio, en particular el permiso de escritura en el directorio, los que determinan quién puede suprimir los ficheros que hay dentro.

4. Cree una máscara restrictiva: usted puede hacer lo que quiere, el grupo sólo tiene acceso a los directorios y puede leer los ficheros, pero los demás no pueden hacer nada.  
La máscara debe dejar pasar todos los permisos del usuario: 0.  
La máscara debe dejar pasar los permisos r y x para los grupos. Sólo w está enmascarado: 2.  
La máscara suprime todos los permisos a los demás: 7.  
Resultado:

```
# umask 027
```

5. Retire el permiso SUID a /usr/bin/passwd y modifique su contraseña. Intente modificar su contraseña. No funciona: passwd debe ser root para modificar los ficheros. Vuelva a poner el permiso s.

```
# chmod u-s /usr/bin/passwd  
# passwd =>Error  
# chmod u+s /usr/bin/passwd
```

# Representación de los discos

Nota previa: las unidades de medida de almacenamiento usadas en este capítulo y en todo el libro usan la representación tradicional, según la regla 1KB = 1024 bytes ( $2^{10}$ ), a no ser que se indique lo contrario.

## 1. Nomenclatura

Este apartado realiza un repaso a los puntos ya vistos en el capítulo Presentación de Linux. En función del tipo de controlador e interfaz en los cuales se conectan los discos, Linux da diferentes nombres a los ficheros especiales que representan discos duros.

Cada disco y cada partición está representado por un archivo especial de tipo bloque.

### a. IDE

Los discos con controladores IDE (también llamados PATA, Parallel Ata o ATAPI) se llaman hdX:

- hda: IDE0, Master
- hdb: IDE0, Slave
- hdc: IDE1, Master
- hdd: IDE1, Slave
- etc.

Contrariamente a lo que se cree, no hay límite al número de controladores IDE, más allá del número de los puertos de extensión de la máquina (slots PCI). Hay muchas tarjetas adicionales y placas base que disponen de cuatro, seis, ocho conectores. En estos casos, los ficheros especiales reciben los nombres hde, hdf, hdg, etc.

Linux entiende que los lectores de CD-Rom, DVD y grabadores son discos IDE y respeta la nomenclatura arriba citada.

Los últimos núcleos de Linux utilizan por defecto un API llamado libata para acceder al conjunto de los discos IDE, SCSI, USB, Firewire, etc. En este caso (puede comprobarlo dirigiéndose a las notas de versión de la distribución), la nomenclatura sigue la de los discos SCSI, que tratamos en el punto siguiente.

### b. SCSI, SATA, USB, FIREWIRE, etc.

Los discos con controladores SCSI, SCA, SAS, FiberChannel, USB, Firewire (y probablemente otras interfaces exóticas, como los lectores ZIP en puerto paralelo) se llaman sdX. La enumeración de los discos sigue el orden de detección de las tarjetas SCSI y de los adaptadores (hosts) asociados, más la adición o supresión manual de otras interfaces de discos duros mediante hotplug.

- sda: primer disco SCSI
- sdb: segundo disco SCSI
- sdc: tercer disco SCSI
- etc.

La norma SCSI marca una diferencia entre los diversos soportes. Así, los lectores de CD-Rom, DVD, HD-DVD, BlueRay y los grabadores asociados no llevan el mismo nombre. Los lectores y grabadores están en srX (sr0, sr1, etc.). También puede encontrar scd0, scd1, etc. Pero suelen ser vínculos simbólicos hacia sr0, sr1, etc.

El comando **lsscsi** permite enumerar los periféricos SCSI.

```
$ lsscsi
[4:0:0:0]    disk    ATA      ST380011A      8.01  /dev/sda
[5:0:0:0]    cd/dvd  LITE-ON  COMBO SOHC-4836V S9C1  /dev/sr0
[31:0:0:0]   disk    USB2.0   Mobile Disk     1.00  /dev/sdb
```

## 2. Casos especiales

### a. Controladores específicos

Algunos controladores no siguen esta nomenclatura. Por ejemplo, es el caso de algunos controladores RAID físicos. Hay que verlo caso por caso. Un controlador Smart Array en un servidor HP coloca sus ficheros de periféricos en /dev/cciss con los nombres cXdYpZ, donde X es el slot, Y el disco y Z la partición...

### b. Virtualización

La representación de discos de sistemas invitados (guests) virtualizados depende del tipo de controlador simulado. La mayoría son de tipo IDE o SCSI, y en ambos casos muy a menudo con libata son vistos como SCSI. Sin embargo, algunos sistemas, como por ejemplo KVM que ofrece paravirtualización, disponen de un controlador específico que presenta los discos con el nombre vdx (virtual disk x):

- vda: primer disco virtualizado
- vdb: segundo disco virtualizado
- etc.

### c. SAN, iSCSI, multipathing

Los discos conectados a través de una SAN (*Storage Area Network*, generalmente con fibra óptica) o mediante iSCSI se ven como discos SCSI y conservan esta nomenclatura. Sin embargo, los sistemas de gestión de rutas múltiples (multipathing) se ubican por debajo, proporcionando otros nombres. Powerpath llamará a los discos emcpowerx (emcpowera, emcpowerb, etc.) mientras que el sistema por defecto de Linux llamará a los mpathx (mpath0, mpath1, etc.) o de cualquier otro modo elegido por el administrador.

# Operaciones de bajo nivel

## 1. Información

El comando **hdparm** permite efectuar un gran número de operaciones directamente sobre los discos duros gestionados por la librería libata, o sea todos los discos SATA, ATA (IDE) y SAS. El comando **sddparm** puede hacer más o menos lo mismo para los discos SCSI. Observe que, a pesar de que los nombres de periféricos de la libata sean idénticos a los del SCSI, es más que probable que muchas opciones de configuración de hdparm no funcionen en discos SCSI. Lo mismo vale para sddparm con los discos SATA o IDE. Los ejemplos que damos a continuación se basan en hdparm.

Para obtener información completa relativa a un disco, utilice los parámetros **-i** o **-I**. El primero recupera la información, desde el núcleo, que se obtiene en el momento del arranque. El segundo interroga directamente al disco. Prefiera **-I** porque da una información muy detallada.

```
# hdparm -I /dev/sda

/dev/sda:

ATA device, with non-removable media
      Model Number:      ST380011A
      Serial Number:    5JVTH798
      Firmware Revision: 8.01
Standards:
      Used: ATA/ATAPI-6 T13 1410D revision 2
      Supported: 6 5 4
Configuration:
      Logical      max      current
      cylinders     16383    16383
      heads         16        16
      sectors/track 63        63
      --
      CHS current addressable sectors:   16514064
      LBA   user addressable sectors:   156301488
      LBA48 user addressable sectors:   156301488
      device size with M = 1024*1024:       76319 MBytes
      device size with M = 1000*1000:       80026 MBytes (80 GB)
Capabilities:
      LBA, IORDY(can be disabled)
      Standby timer values: spec'd by Standard, no device specific
minimum
      R/W multiple sector transfer: Max = 16 Current = 16
      Recommended acoustic management value: 128, current value: 0
      DMA: mdma0 mdma1 mdma2 udma0 udma1 udma2 udma3 udma4 *udma5
          Cycle time: min=120ns recommended=120ns
      PIO: pio0 pio1 pio2 pio3 pio4
          Cycle time: no flow control=240ns  IORDY flow control=120ns
Commands/features:
      Enabled Supported:
          * SMART feature set
          * Security Mode feature set
          * Power Management feature set
          * Write cache
          * Look-ahead
          * Host Protected Area feature set
          * WRITE_BUFFER command
          * READ_BUFFER command
          * DOWNLOAD_MICROCODE
          * SET_MAX security extension
          * 48-bit Address feature set
```

```
*      Device Configuration Overlay feature set
*      Mandatory FLUSH_CACHE
*      FLUSH_CACHE_EXT
*      SMART error logging
*      SMART self-test
*      General Purpose Logging feature set
Time Limited Commands (TLC) feature set
Command Completion Time Limit (CCTL)
```

#### Security:

```
Master password revision code = 65534
supported
not enabled
not locked
not frozen
not expired: security count
not supported: enhanced erase
```

#### HW reset results:

```
CBLID- above Vih
Device num = 0 determined by CSEL
```

Checksum: correct

## 2. Modificación de los valores

Se puede modificar varios parámetros de los discos. Sin embargo, ¡cuidado! Algunas opciones de **hdparm** pueden resultar peligrosas tanto para los datos contenidos en el disco como para el propio disco. La mayoría de los parámetros son de lectura y escritura. Si no se especifica ningún valor, **hdparm** muestra el estado del disco (o del bus) para este comando. A continuación le presentamos algunos ejemplos de opciones interesantes.

- **-c**: anchura del bus de transferencia EIDE en 16 o 32 bits. 0=16, 1=32, 3=32 compatible.
- **-d**: utilización del DMA. 0=no DMA, 1=DMA activado.
- **-x**: modifica el modo DMA (mdma0 mdma1 mdma2 udma0 udma1 udma2 udma3 udma4 udma5). Puede utilizar cualquiera de los modos anteriores o valores numéricos: 32+n para los modos mdma (n varía de 0 a 2) y 64+n para los modos udma.
- **-c**: modo de ahorro de energía en el disco (unknown, active/idle, standby, sleeping). Se puede modificar el estado con **-S**, **-y**, **-Y** y **-Z**.
- **-g**: muestra la geometría del disco.
- **-m**: indica o modifica el estado del Automatic Acoustic Management (AAM). 0=off, 128=quiet y 254=fast. No todos los discos lo soportan.
- **-r**: pasa el disco en sólo lectura.
- **-t**: bench de lectura de la caché del disco, ideal para probar la eficacia de transferencia entre Linux y la caché del disco. Hay que volver a ejecutar el comando dos o tres veces.
- **-t**: bench de lectura del disco, fuera de la caché. Mismas observaciones que para la opción anterior.

Así, el comando siguiente pasa el bus de transferencia a 32 bits, activa el modo DMA en modo Ultra DMA 5 para el disco sda:

```
# hdparm -c1 -d3 -X udma5 /dev/sda
```

Le mostramos a continuación otros ejemplos:

```
# hdparm -c /dev/sda
```

```
/dev/sda:  
  IO_support      =  0 (default 16-bit)  
  
# hdparm -C /dev/sda  
  
/dev/sda:  
  drive state is:  active/idle  
  
# hdparm -g /dev/sda  
  
/dev/sda:  
  geometry      = 9729/255/63, sectors = 156301488, start = 0  
p64p17bicb3:/etc/cups #  
  
# hdparm -T /dev/sda  
  
/dev/sda:  
  Timing cached reads:   1320 MB in  2.00 seconds = 660.30 MB/sec  
  
# hdparm -t /dev/sda  
  
/dev/sda:  
  Timing buffered disk reads:  168 MB in  3.03 seconds =  55.49 MB/sec
```

# Elegir un sistema de ficheros

## 1. Fundamentos

### a. Definición de sistema de ficheros

La acción de «formatear» un disco, una llave USB o cualquier soporte de datos consiste únicamente en crear en un soporte de memoria secundaria la organización lógica que permite colocar datos en él. Apenas se utiliza la palabra «formateo» en Linux, excepto para explicar el fundamento teórico a las personas procedentes de otros lados. Se habla de sistema de ficheros, que representa a la vez la organización lógica de los soportes tanto a un nivel inferior como a un nivel de usuario.

No se escribe la información en los discos de cualquier manera. Se requiere una mínima organización para colocar en ellos tanto la información relativa a los ficheros como los datos almacenados. Es el sistema de ficheros (y los controladores asociados) el que define esta organización. Si bien los fundamentos organizativos suelen ser los mismos en los diferentes sistemas de ficheros presentes soportados por Linux, las implementaciones y organizaciones lógicas de los datos en el disco varían bastante de uno a otro. De esta manera, no hay un único tipo de sistema de ficheros, sino varios, puestos a disposición del usuario, el administrador o el ingeniero.

El principio básico es asociar un nombre de fichero con su contenido y autorizar su acceso: creación, modificación, supresión, desplazamiento, apertura, lectura, escritura, cierre. Conforme a este principio, el sistema de ficheros debe gestionar lo que deriva de ello: mecanismos de protección de los accesos (permisos, propietarios), accesos concurrentes, etc.

### b. Representación

Además de la organización y el almacenamiento de la información y datos en los ficheros, el sistema de ficheros debe facilitar al usuario una visión estructurada de sus datos, que permite distinguirlos, encontrarlos, tratarlos y trabajar con ellos, por ejemplo en forma de ficheros dentro de una estructura de directorios con los comandos asociados. Asimismo, cada sistema de ficheros debe proporcionar lo necesario para que los programas puedan acceder a él.

Un sistema de ficheros Unix se organiza en forma de un árbol de directorios y subdirectorios desde una raíz común. Es una estructura en árbol. Cada directorio forma parte de una organización y propone, a su vez, una organización: el sistema de ficheros dispone de una jerarquía ordenada. Se puede repartir la propia estructura en árbol entre varios soportes y sistemas de ficheros.

### c. Los metadatos

Las propiedades de un fichero se denominan metadatos, aunque en Linux recibe el nombre de inodo. Los inodos o metadatos de todo fichero están ubicados en un lugar especial del soporte de almacenamiento. El contenido de los metadatos cambia de un sistema de ficheros a otro. Sin embargo, en Linux, encontramos siempre este contenido:

- los permisos;
- las últimas fechas de acceso y modificación;
- el propietario y el grupo;
- el tamaño;
- el número de bloques utilizados;

- el tipo de ficheros;
- el contador de vínculos;
- un árbol de direcciones de bloques de datos.

#### **d. Los nombres de los ficheros**

Los nombres pueden tener una longitud de 255 caracteres. La extensión no es relevante como componente del sistema de ficheros a diferencia de lo que ocurre en Windows. Se determina el tipo de fichero según su contenido, en particular el de los primeros bytes, que permiten determinar el tipo MIME. El comando **file** procede de esta manera. La extensión es un mero componente del nombre del fichero, incluida en los 255 caracteres. Ante todo, es útil para que el usuario diferencie rápidamente los ficheros entre sí.

Los nombres de los ficheros Unix no se establecen en los metadatos, sino en una tabla de catálogo. Gracias a ello es posible dar varios nombres a un mismo fichero.

#### **e. El fichero de diario**

Los actuales sistemas de ficheros disponen a menudo de mecanismos que permiten garantizar lo mejor posible la integridad de los datos. El sistema más habitual es el journaling (es un anglicismo). El sistema de ficheros mantiene al día un registro, el diario, en general de un tamaño dado y circular (la nueva información termina por chafar la antigua) en el cual registra todos los cambios antes de efectuarlos realmente. En caso de parada abrupta, el sistema repasa las grabaciones del diario y comprueba si se realizaron las operaciones. Si es necesario las vuelve a ejecutar. El diario contiene operaciones atómicas (n operaciones indivisibles) y, por lo tanto, incluso si éste no es completo, se asegura la coherencia de los datos o bien completando el diario cuando es posible, o bien dando un paso atrás. Por lo tanto, la reparación es mucho más fiable y rápida que en un sistema de archivos tradicional.

## **2. Los sistemas de ficheros en Linux**

#### **a. ext2**

Se considera que el «second extended filesystem» ext2 era el sistema de ficheros histórico de Linux, aunque éste utilizaba el MinixFS inicialmente. La primera versión, llamada ext (extended filesystem), aunque corría los defectos de minix, tenía unos límites que impedían que fuera en un verdadero sistema de ficheros Unix. Ext2 es, por lo tanto, el primer sistema de ficheros desarrollado específicamente para Linux, de un nivel de producción y conforme a las normas Unix (se habla de nivel de producción para indicar que un sistema cualquiera responde a los criterios de puesta en producción -uso real- en una empresa). Previsto desde el principio para soportar el aumento de funcionalidades, se sigue utilizando y mejorando desde 1993. Ext2 no es «journaling».

Aunque disponga de un sucesor (ext3), se sigue utilizando e incluso aconsejando en algunos casos. Es rápido y necesita menos escritura que los demás y, por lo tanto, conlleva menos desgaste de los soportes de almacenamiento, en particular los discos SSD, las claves USB o las tarjetas memoria. A veces estos soportes sólo pueden soportar un número restringido de ciclos de lectura/escritura...

Los ficheros pueden tener un tamaño de hasta 2 TB (2048 GB), mientras que una partición puede alcanzar 32 TB, incluso 128 TB, según el tamaño de los bloques y la arquitectura.

#### **b. ext3**

El *third extended filesystem* ext3 es el sucesor de ext2 desde 1999. Cuenta con un diario. Sobre todo, es

totalmente compatible con ext2. El fichero de diario es una extensión de ext2. Es posible utilizar un sistema de ficheros ext3 como ext2, con los mismos comandos, las mismas operaciones. Es posible transformar en unos segundos un sistema ext2 en ext3, y viceversa. Es uno de los sistemas de ficheros de elección para Linux, y el más utilizado dada su flexibilidad.

Como para ext2, el tamaño máximo de los ficheros es de 2 TB, y el de una partición, de 32 TB, siguiendo las mismas restricciones.

### c. ext4

El conocido como *fourth extended filesystem* ext4 es el sucesor de ext3. Estable desde 2008, es el sistema de archivos predeterminado de un gran número de distribuciones desde 2010. Durante su fase de puesta a punto se le conocía como ext4dev. Es compatible con ext3 y, aunque es avanzado, es considerado como el sistema de archivos de transición ante la llegada de **btrfs**. Como ext3, tiene diario. Añade nuevas funcionalidades, especialmente el concepto de extents, una técnica de preasignación de bloques en zonas contiguas que permite limitar la fragmentación de archivos y del disco y, por tanto, mejora el rendimiento. La rapidez de las comprobaciones del sistema de archivos ha sido revisada y ahora es incluso más rápido que ext3.

Un sistema de archivos ext3 puede montarse como ext4. Sin embargo, una partición ext4 sólo puede montarse como ext3 (o ext2) si el sistema de archivos ha sido creado sin el uso de extents.

El tamaño máximo de los archivos es de 16 TB y el de una partición de 1024 PB (Petabytes), es decir 1 EB (1 Exabyte).

### d. btrfs

El sistema de archivos btrfs, que se pronuncia “butterFS”, es un sistema de archivos considerado, mientras se escribe este libro, todavía experimental y es el llamado a ser el nuevo sistema de archivos predeterminado de Linux. Inicialmente desarrollado por Oracle, btrfs está en desarrollo continuo. Algunos lo usan ya habitualmente. Utiliza el concepto de extents como ext4 y el de subvolúmenes: la creación de árboles separados en el interior de un sistema de archivos, con opciones propias en cada árbol (permisos, cuotas, etc.). Este método también permite la creación de instantáneas (*snapshots*) de un árbol, la imagen exacta del árbol en un instante determinado, permitiendo comprobar las modificaciones o volver al estado inicial, como con LVM. btrfs establece un sistema de protección de datos y de metadatos, especialmente con checksums que permiten averiguar si un archivo se ha corrompido.

Otra ventaja de btrfs es que permite redimensionar el sistema de archivos en caliente, incluyendo la posibilidad de reducirlo, característica que no disponen sistemas como ext2/3/4. Además, un sistema de archivos btrfs se puede organizar en varios volúmenes (discos o particiones), lo que le hace parecerse a soluciones del tipo LVM. Por lo tanto, btrfs representa el estado del arte como sistema de archivos para equipos de cliente y servidores

### e. reiserfs

reiserfs fue el primer sistema de ficheros integrado en Linux, antes incluso de ext3. Su fuerza reside, (además de en su fichero de diario) en la organización indexada de las entradas de los directorios (las tablas de catálogos que contienen las asociaciones inodos/ficheros) y el manejo de los ficheros de pequeño tamaño. Sus resultados son excepcionales cuando hay miles de ficheros, de poco a mediano volumen. Puede ser redimensionado en un momento. El problema más relevante es que, con ficheros muy grandes, es lento.

Los ficheros pueden alcanzar 8 TB, y las particiones, 16 TB. Los nombres de ficheros pueden tener 4032 caracteres, pero Linux los limita a 255 caracteres (más precisamente con el **VFS**, *Virtual Filesystem Switch*).

A pesar de las evidentes mejoras que presenta, reiserfs es mucho menos utilizado. La razón estriba en que no es posible convertir un sistema de ficheros ext2/ext3 en reiserfs ni viceversa, y eso le perjudica, ya que ext2/ext3 es omnipresente. reiserfs y su sucesor reiserfs4 han caído un poco en desuso debido a los

problemas judiciales de su creador, en la cárcel durante varios años, sin ofrecer, por tanto, el mantenimiento de su sistema de archivos.

## f. xfs

xfs es el más antiguo de los sistemas de ficheros transaccional en Linux, con fecha de 1993. Creado por Silicon Graphics (sgi), fue llevado a Linux en 2000. Además de sus capacidades de almacenamiento aún impensables a día de hoy, tiene un sistema de diario transaccional muy eficaz y mecanismos avanzados, como la desfragmentación en línea (en caliente y al mismo tiempo que la escritura), la capacidad de efectuar unos snapshots (fijar el estado de un sistema de ficheros en un instante t para restaurarlo más tarde), el dimensionamiento en caliente, la reserva de ancho de banda habilitada en comunicaciones para las entradas y salidas, etc.

El tamaño máximo (teórico, porque aún hoy nadie ha creado ninguno así de grande) de los ficheros es de 8 EB (Exabytes). 1 EB vale 1024 PB (Petabyte), por lo tanto 1048576 TB, o sea, convertido a una unidad más comprensible, unos 100.000 millones de DVD. La partición puede alcanzar 16 EB, o sea, la capacidad máxima de un controlador en 64 bits. En sistemas de 32 bits, se «limitan» los tamaños a 16 TB.

El uso de xfs no está aún muy extendido en Linux, quizás por culpa de su supuesta complejidad para una configuración avanzada, pero también porque Red Hat no tiene soporte oficial para este sistema de ficheros en sus RHEL.

## g. vfat

**vfat** (*Virtual File Allocation Table*) es un término genérico que agrupa las diferentes versiones de FAT que soportan los nombres de ficheros largos (255 caracteres) en Windows. Se conservan estos sistemas de ficheros y se siguen utilizando por razones a la vez históricas y prácticas. La mayoría de los soportes removibles, discos externos, claves USB y lectores MP3 utilizan un sistema de ficheros de este tipo, debido a que:

- Se trata de un sistema de ficheros adaptado a los pequeños volúmenes.
- Es un sistema de ficheros simple a implementar, ideal para lectores multimedia.
- Compatibilidad entre diversas plataformas (Windows, Linux, BSD, Mac OS, etc.).

vfat padece defectos inherentes a su concepción:

- Se almacena el conjunto de la información dentro de una tabla única, incluyendo el nombre del fichero y cada dirección y longitud de los bloques (llamados clústers) que componen los datos del fichero.
- De hecho, FAT intenta agrupar los datos de un fichero en el mayor número de clústers contiguos del soporte. En caso de que se hagan muchas modificaciones (añadido, supresión, etc.), el sistema se ve fuertemente fragmentado.
- Por otro lado, cuanto más grande es el soporte, más lento va FAT, ya que debe comprobar toda la tabla FAT para encontrar clústers disponibles.
- La gestión de nombres largos es puro bricolaje para muchas personas, ya que FAT continúa asegurando la compatibilidad (aún hoy en día) con los nombres cortos en 8.3.
- A diferencia de los otros sistemas de ficheros de Unix, Linux o Windows recientes, FAT no gestiona ningún atributo extendido, en particular ninguna noción de los permisos y propietarios.
- Se limita FAT a un tamaño de ficheros de 4 GB. Se deben cortar los ficheros más grandes (base de datos, archivos, imágenes, ISO de DVD, etc.) en consecuencia y los programas (captura de audio / vídeo, sgbd, etc.) deben gestionar esta limitación.

Aunque VFAT sea todavía el sistema de archivos predeterminado de muchos lectores multimedia, smartphones, pendrives USB, tarjetas de memoria y dispositivos de fotografía digital, cada vez pierde más protagonismo en relación a NTFS en el caso de los soportes de intercambio y ext3/ext4 (por ejemplo en Android). La llegada de la alta definición amenaza con la desaparición de VFAT debido a que su tamaño de archivo máximo es de 4 GB.

Linux gestiona perfectamente VFAT. Pero su uso en soportes compartidos entre Windows y Linux tiene cada vez menos razón de ser, ya que la existencia de ntfs3g permite utilizar soportes que contienen un sistema de ficheros NTFS de manera nativa.

# Particionamiento

## 1. Particionamiento lógico

En este manual consideraremos soporte de almacenamiento a cualquiera de tipo magnético o de memoria como, por ejemplo, discos duros, SSD, tarjeta memoria, etc. Es decir, todo lo que se pueda parecer a un disco duro según la visión clásica: un espacio de datos que se puede seccionar en varias entidades lógicas e independientes y que dispone cada una de su propio sistema de ficheros.

Se puede considerar un disco como una larga banda de espacio de almacenamiento cortado en casillas que pueden contener una cantidad determinada de información. Se puede utilizar el disco tal cual, como espacio de almacenamiento. Nada impide crear un sistema de ficheros en un disco sin pasar por la etapa de particionamiento. Sin embargo, resulta importante dar una organización lógica a este espacio y a los sistemas de ficheros que va a contener, aunque se trate sólo de separar los datos (los ficheros de datos) de los tratamientos (los programas que los utilizan y el sistema).

El particionamiento consiste en una división lógica del disco. Se fracciona el disco físico, real, en varios discos virtuales, lógicos: las particiones. Se ve cada partición como un disco independiente que contiene su propio sistema de ficheros.

## 2. Organización de un disco

### a. MBR

El primer sector es el **MBR**, *Master Boot Record* o zona de arranque. Con un tamaño de 512 bytes, contiene en sus primeros 446 bytes una rutina (un programa) de arranque destinado a ejecutar el sistema operativo desde la partición activa, o desde el gestor de arranque (bootloader). Los 6 últimos bytes de este bloque, opcionales, son usados por Windows para 'firmar' el disco con 4 bytes. A continuación, 2 bytes nulos, y más adelante, los 64 bytes que contienen la tabla de las cuatro particiones primarias. El conjunto termina con una firma 0xAA55 en dos bytes.



### b. Las particiones

Una partición es una división lógica del disco. Existen tres tipos:

- Las particiones primarias, en un total de cuatro, son las descritas en el MBR.
- Las particiones extendidas (primarias extendidas), una sola por disco (aunque en teoría sea posible crear particiones extendidas dentro de otras particiones extendidas).
- Las particiones lógicas.

Un disco puede contener hasta 63 particiones en IDE, 15 en SCSI (es un límite de la implementación oficial del SCSI) o mediante la librería libata. El límite actual es de 15 particiones para todos los discos con los últimos núcleos y la API libata. Sin embargo, algunas distribuciones permiten utilizar la antigua API (PATA)

para volver al antiguo sistema.

Observe que se trata efectivamente de un límite por disco, y no del número total de particiones gestionadas por el sistema.

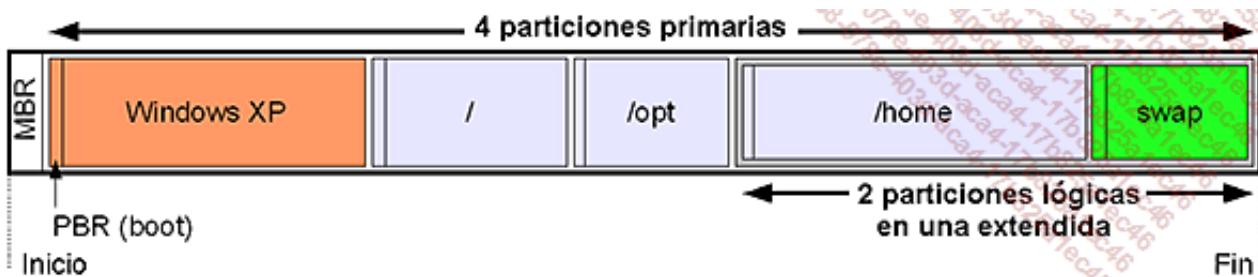
- Para superar el límite de las 15 particiones, es posible utilizar el «device mapper» de Linux, que requiere en particular la gestión del **LVM** (*Logical Volume Management*). El LVM permite agrupar varios discos (Physical Volumes) en una sola unidad (Volume Group) vista por el sistema como un enorme disco único que usted puede cortar en particiones (Logical Volumes) sin limitación de número. Además, puede añadir discos en el grupo después, aumentar o reducir el tamaño de las particiones sin preocuparse de su ubicación física real...

Se numeran las particiones de 1 a n (15 o 63). Una partición con un valor superior o igual a 5 indica que se trata obligatoriamente de una partición lógica. Como sólo puede haber cuatro particiones primarias, se suele crear la última (la 4) como extendida:

- Particiones 1 a 3: primarias
- Partición 4: extendida
- Particiones 5 a n: lógicas

El número de la partición aparece a continuación del nombre del fichero periférico de disco:

- hda1: primera partición primaria del primer disco IDE;
- hdb5: quinta partición, primera partición lógica del segundo disco IDE;
- sda3: tercera partición primaria del primer disco SCSI / libata;
- sdc8: octava partición, o sea, cuarta partición lógica del tercer disco SCSI/libata.



Descripción esquemática de un disco

### c. EBR

Como cada partición extendida debe describir las particiones lógicas que contiene, debe disponer también de una tabla de partición. El **EBR** (*Extended Boot Record*) retoma la estructura del MBR, excepto por que sólo hay dos grabaciones posibles en la tabla de las particiones. La primera indica efectivamente la posición y el tamaño de una partición lógica, mientras que la segunda está vacía si es la única partición lógica, o apunta a otro EBR. Por lo tanto, puede haber varios EBR en una partición extendida.

- Los EBR forman una lista encadenada, la segunda entrada de partición apuntando al EBR siguiente.
- Sólo hay una partición lógica descrita por EBR.

### d. PBR

El **PBR** (*Partition Boot Record*), también llamado **VBR** (*Volume Boot Record*) o Partition Boot Sector es el primer sector de cada partición primaria o lógica. Puede contener una rutina de inicio de un sistema operativo, un cargador de inicio o incluso nada si la partición no tiene como misión iniciar un sistema operativo. Cuando el MBR no contiene rutinas, la BIOS intenta iniciar y ejecutar la rutina del PBR de la partición marcada como activa.

## e. Tipos de particiones

Cada partición dispone de un tipo que permite determinar su contenido. Es un identificador numérico codificado en un byte generalmente presentado en hexadecimal. Parece importante facilitar una lista aquí para que usted pueda entender correctamente la finalidad de este identificador. Los valores más comunes están en negrita.

1	FAT12	24	NEC DOS	81	Minix / old Lin	bf	Solaris
2	XENIX root sec	39	Plan 9	82	Linux swap / So	c1	DRDOS/
3	XENIX usr sec	3c	PartitionMagic	<b>83</b>	Linux	c4	DRDOS/
4	FAT16 <32M sec	40	Venix 80286	84	OS/2 hidden C:	c6	DRDOS/
5	Extended	41	PPC PReP Boot	85	Linux extended	c7	Syrinx
<b>6</b>	<b>FAT16</b>	42	SFS	86	NTFS volume set	da	Non-FS
data							
<b>7</b>	<b>HPFS/NTFS</b>	4d	QNX4.x	87	NTFS volume set	db	CP/M /
CTOS / .							
8	AIX Utility	4e	QNX4.x 2nd part	88	Linux plein tex	de	Dell
9	AIX bootable	4f	QNX4.x 3rd part	<b>8e</b>	Linux LVM	df	BootIt
a	OS/2 Boot Manag	50	OnTrack DM	93	Amoeba	e1	DOS
access							
<b>b</b>	<b>W95 FAT32</b>	51	OnTrack DM6 Aux	94	Amoeba BBT	e3	DOS R/O
<b>c</b>	<b>W95 FAT32 (LBA)</b>	52	CP/M	9f	BSD/OS	e4	Speed-
Stor							
<b>e</b>	<b>W95 FAT16 (LBA)</b>	53	OnTrack DM6 Aux	a0	IBM Thinkpad hi	eb	BeOS fs
<b>f</b>	<b>W95 Etendu (LBA</b>	54	OnTrackDM6	a5	FreeBSD	ee	EFI GPT
10	OPUS	55	EZ-Drive	a6	OpenBSD	ef	EFI
(FAT-12/16/							
11	Hidden FAT12 PA-RISC b	56	Golden Bow	a7	NeXTSTEP	f0	Linux/
12	Compaq diagnost	5c	Priam Edisk	a8	UFS Darwin	f1	Speed-
Stor							
14	Hidden FAT16 <3	61	SpeedStor	a9	NetBSD	f4	Speed-
Stor							
16	Hidden FAT16 secondary	63	GNU HURD or Sys ab	Amorce	Darwin	f2	DOS
17	Hidden HPFS/NTF	64	Novell Netware	b7	BSDI fs	<b>fd</b>	Linux
raid auto							
18	AST SmartSleep	65	Novell Netware	b8	BSDI swap	fe	LANstep
1b	Hidden W95 FAT3	70	DiskSecure Mult	bb	Boot Wizard hid	ff	BBT
1c	Hidden W95 FAT3	75	PC/IX				

Como el tipo de partición debería reflejar el sistema de ficheros que contiene, una partición de tipo 0x0c debería contener un sistema de ficheros de tipo FAT32 LBA (discos grandes). Una partición de tipo 0x83 debería contener un sistema de ficheros Linux. Pero ¿cuál? Ya ha visto que existen varios...

Observe el predominio de los tipos de partición para Windows. Windows se basa esencialmente en este tipo para determinar su contenido. Nada impide crear una partición de tipo Linux y colocar en ella un sistema de ficheros FAT32. Sin embargo, si hace esto, Windows no reconocerá la partición (considerada de tipo desconocido) y usted no podrá acceder al contenido.

Linux reconocerá en general (hay una pocas excepciones) el contenido de una partición por el sistema de

ficheros que reside en ella. Usted puede crear un sistema de ficheros ext3 en una partición de tipo 0x0e y constatar que todo funciona. El tipo 0x83 puede acoger cualquier sistema de ficheros Linux: ext2, ext3, reiserfs, jfs, xfs... Sin embargo, por razones de compatibilidad, tenga cuidado en respetar la asociación tipo de partición ↔ sistema de ficheros y ser coherente.

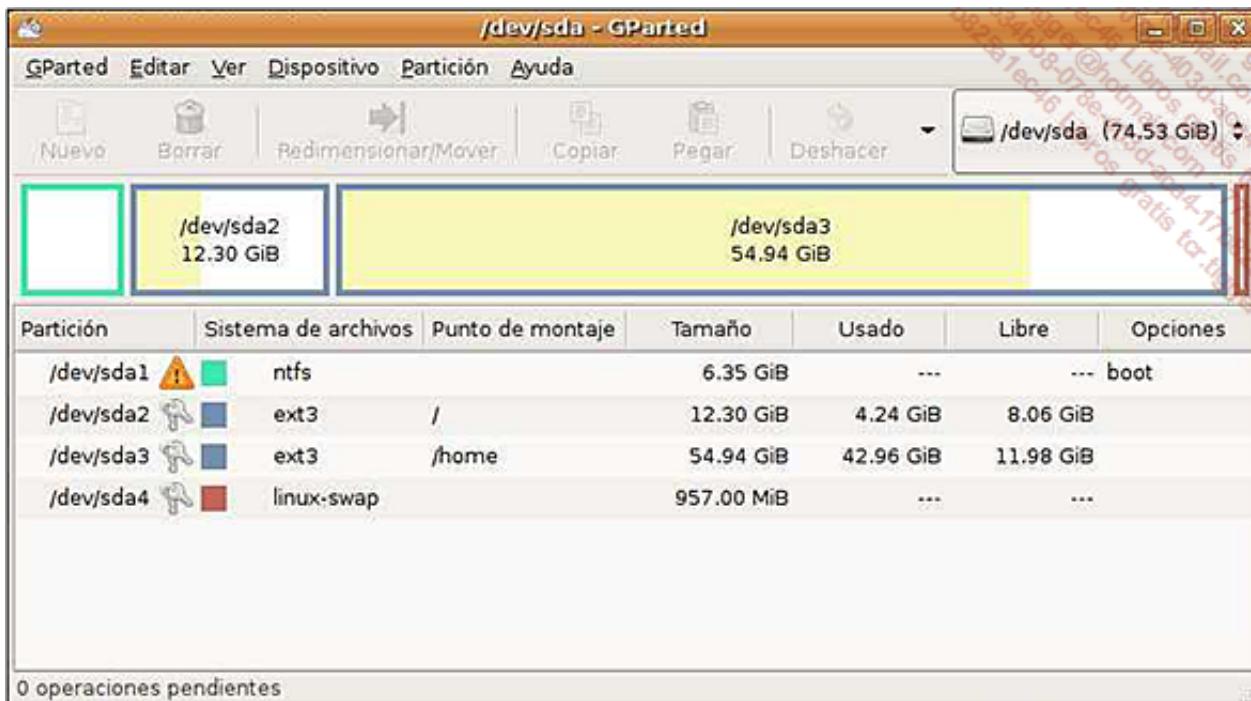
### 3. Manejar las particiones

#### a. Herramientas de gestión de particiones

Las herramientas **fdisk**, **cfdisk**, **sfdisk** o también **parted**, sin contar con las herramientas gráficas disponibles durante la instalación o en los paneles de configuración, permiten manejar las particiones.

- **fdisk** es la más antigua y más utilizada de las herramientas de particionamiento. No tiene relación con el fdisk de Microsoft. Se basa en menús y atajos textuales.
- **cfdisk** es un poco más «visual» y se utiliza con las flechas direccionales. Permite las mismas operaciones que fdisk y es de fácil manejo.
- **sfdisk** funciona, opcionalmente, de forma interactiva. Es bastante complicada, pero más precisa.
- **parted** permite operaciones muy avanzadas en las particiones, como, por ejemplo, su redimensionamiento. Presenta una interfaz interactiva (intérprete de comandos) que atiende a scripts. Pero, además, hay en el mercado interfaces gráficas de parted, como qtparted o gparted.

En la captura siguiente puede ver a **gparted** en acción.



*gparted, un editor de particiones gráfico*

#### b. Manejar las particiones

##### Listar

Los administradores e ingenieros de sistemas suelen utilizar la herramienta **fdisk**, que es a la vez la más antigua y más estándar. Hay que tener privilegios de root para poder usar fdisk.

**fdisk [-l] [disco]**

Cada parámetro es opcional. Iniciado tal cual, **fdisk** se sitúa en el primer disco del sistema. El parámetro -l permite listar las particiones del disco dado o de todos los discos. La información obtenida es la misma que en modo interactivo con la entrada p (print) del menú.

```
# fdisk -l /dev/sda

Disco /dev/sda: 164,6 Gb, 164696555520 bytes
255 heads, 63 sectors/track, 20023 cylinders
Units = cilindros of 16065 * 512 = 8225280 bytes
Disk identifier: 0x000c02ae

Periférico Arranque Principio Fin Bloques Id Sistema
/dev/sda1 1 5222 41945683+ 7 HPFS/NTFS
/dev/sda2 * 5223 5288 530145 83 Linux
/dev/sda3 5289 10510 41945715 83 Linux
/dev/sda4 10511 20023 76413172+ f W95 Extendido (LBA)
/dev/sda5 10511 10772 2104483+ 82 Linux swap / Solaris
/dev/sda6 10773 20023 74308626 83 Linux
```

Los campos hablan por sí mismos. Fíjese en que la partición sda4 es la extendida. Tiene como tipo 0x0f, pero cualquier tipo extendido hubiese funcionado: se consideran los tipos 0x05 y 0x85 como idénticos. Sin embargo, es posible que Windows no reconozca estos tipos ni, por lo tanto, las particiones lógicas contenidas en ellos, en particular las de un disco LBA. Por eso, la mayoría de las herramientas de particionamiento de las distribuciones Linux prefieren el tipo asociado a Windows.

Sólo se tienen en cuenta las operaciones efectuadas con **fdisk** al final, una vez que haya guardado usted sus modificaciones, y no al mismo tiempo. Si piensa que se ha equivocado, no dude en salir sin guardar o en hacer un [Ctrl] C. Se perderán sus modificaciones, pero habrá salvado sus particiones.

### **Lista de particiones**

El ejemplo siguiente se basa en un disco reconocido por el sistema como /dev/sdb y que no contiene ninguna partición. El objetivo es crear tres particiones: una primaria, una extendida y una lógica.

- Ejecute **fdisk** con el disco como argumento. No tenga en cuenta las primeras líneas visualizadas, excepto si indican un error.

```
# fdisk /dev/sdb
...
Comando (m para la ayuda) :
```

- Verifique primero la existencia de particiones con la tecla p (print), luego [Intro].

```
Comando (m para la ayuda) : p

Disco /dev/sdb: 4026 Mb, 4026531840 bytes
64 heads, 62 sectors/track, 1981 cylinders
Units = cilindros of 3968 * 512 = 2031616 bytes
Disk identifier: 0x0003ed63

Periférico Arranque Principio Fin Bloques Id Sistema
/dev/sdb1 * 1 1981 3930273 c W95 FAT32 (LBA)
```

## **Suprimir**

Para suprimir una partición, utilice la tecla d (delete); luego, si hay varias particiones, el número de partición (sdbX, siendo X el número). Si hay una única partición, se coge por defecto.

```
Comando (m para la ayuda): d  
Partición seleccionada 1
```

## **Crear**

Para crear una partición, utilice la tecla n (new). Luego debe elegir el tipo de partición: primaria o extendida.

```
Comando (m para la ayuda): n  
Acción de comando  
e   extendida  
p   partición primaria (1-4)
```

Para esta primera partición, seleccione una partición primaria con la tecla p (que significa «primary»-esta vez).

- Como el MBR contiene cuatro entradas, puede elegir el número de partición a crear. Es perfectamente posible crear la partición sdb2 antes de la sdb1. Aquí, teclee **1**.
- El primer cilindro corresponde a la posición de principio de su partición. Por defecto, fdisk se coloca en el primer cilindro disponible desde el principio del disco. Es perfectamente posible crear particiones primarias en la mitad de un disco. Seleccione aquí el valor por defecto (1) pulsando [Intro].
- Finalmente, elija el tamaño de la partición. Es preferible utilizar una unidad legible, como los KB o MB. Por ejemplo, para una partición de 1 GB, es decir, de 1024 MB, use **+1024M** y pulse [Intro]. Ahora se ha definido la partición.

```
Comando (m para la ayuda): n  
Acción de comando  
e   extendida  
p   partición primaria (1-4)  
p  
Número de partición (1-4): 1  
Primer cilindro (1-1981, por defecto 1):  
Utilización del valor por defecto 1  
Último cilindro o + tamaño o +tamañoM o +tamañoK (1-1981, por defecto 1981): +1024M
```

- Verifique el estado de la partición (p).

```
Comando (m para la ayuda): p  
...  
Periférico Arranque Principio Fin Bloques Id Sistema  
/dev/sdb1 1 505 1001889 83 Linux
```

## **Guardar**

Salga de fdisk guardando su tabla de las particiones con la tecla w (write). Fdisk escribe la nueva tabla de las particiones en el MBR o los EBR. Puede que el sistema le muestre alertas que nosotros aquí le indicamos en negrita.

```
Comando (m para la ayuda): w  
¡Se alteró la tabla de particiones!
```

Llamada de ioctl() para volver a leer la tabla de particiones.

**ADVERTENCIA: la relectura de la tabla de particiones fracasó con el error 16: Periférico o recurso ocupado.**

**El kernel va a seguir utilizando la antigua tabla.**

Se utilizará la nueva tabla durante el próximo reinicio.

Sincronización de los discos.

Este mensaje significa que, como el disco está en uso, Linux no puede volcar la nueva tabla ni, por lo tanto, crear las nuevas particiones. El comando siguiente lo puede confirmar. Observe que la última línea debería mostrarle su nueva partición. Pero no es así.

```
# cat /proc/particiones | grep sdb
8 16 3932160 sdb
```

### **Forzar la sincronización**

Para corregir este último problema y forzar al núcleo a leer de nuevo la tabla de las particiones, tiene a su disposición dos comandos. El primero es **blockdev** con el parámetro --rereadpt (re-read partition table).

```
# blockdev --rereadpt /dev/sdb
```

El segundo es **partprobe**, disponible solamente si parted está instalado. Puede probarlo si blockdev no funciona. Por defecto, vuelve a leer las tablas de todas las particiones, pero le puede especificar como argumento un disco en concreto.

```
# partprobe /dev/sdb
```

Verifique si se reconoce la partición.

```
# cat /proc/particiones | grep sdb
8 16 3932160 sdb
8 17 1001889 sdb1
```

Ahora puede utilizar su nueva partición y añadirle un sistema de ficheros. Cree ahora una partición extendida, luego una partición lógica. Observe que se crea una partición extendida o lógica de la misma manera que las otras. Una partición extendida no tiene por qué ser la última de las primarias. También puede ser la segunda, por ejemplo. Si no ha utilizado todo el tamaño para crearla, puede completar más adelante la creación de particiones primarias.

 Se pueden usar los comandos de parted como **mkpart**, **rm** o **resize** para trabajar con las particiones. El comando **partx** también permite notificar al núcleo las modificaciones realizadas a las tablas de partición.

### **Modificar el tipo**

Como la modificación del tipo de una partición no implica la modificación de su tamaño, puede hacerlo en cualquier momento. Le presentamos a continuación el modo de proceder para pasar del tipo por defecto Linux al tipo FAT para que Windows reconozca la partición.

- Ejecute **fdisk** y pulse t (tipo).
- Seleccione la partición, la 5 para la primera partición lógica.

Puede visualizar todos los tipos pulsando L, que le facilita la misma lista mencionada más arriba. Utilice el tipo c W95 FAT32 (LBA) para estar tranquilo.

- Guarde con w.

```
Comando (m para la ayuda): t
Número de partición (1-5): 5
Código Hex (teclear L para listar los códigos): c
Tipo de partición del sistema modificado de 5 a c (W95 FAT32 (LBA))

Comando (m para la ayuda): p

Disco /dev/sdb: 4026 Mb, 4026531840 bytes
64 heads, 62 sectors/track, 1981 cylinders
Units = cilindros of 3968 * 512 = 2031616 bytes
Disk identifier: 0x0003ed63

Periférico Arranque Principio Fin Bloques Id Sistema
/dev/sdb1 1 505 1001889 83 Linux
/dev/sdb2 506 1981 2928384 5 Extended
/dev/sdb5 506 1981 2928353 c W95 FAT32 (LBA)

Comando (m para la ayuda): w
```

Los tipos de particiones más utilizados para Linux son los siguientes:

- **83**: Partición de tipo Linux (datos)
- **82**: Partición de tipo swap
- **fd**: Partición de tipo RAID
- **8e**: Partición de tipo LVM

# Manejar los sistemas de ficheros

## 1. Definición básica

### a. Bloque

El bloque es la unidad básica, atómica, de almacenamiento del sistema de ficheros. Un fichero ocupa siempre un número entero de bloques. Así, si el tamaño de un fichero es de un byte y el bloque en el que está tiene un tamaño de 4096 bytes, se desperdician 4095 bytes. De esta manera, es posible llenar un sistema de ficheros donde n ficheros de 1 bytes, donde n representa el número de bloques, mientras que el volumen total de los datos sólo es de n bytes!

Supongamos un disco que contiene 102.400 bloques de 4096 bytes. Su tamaño total es de 400 MB. Supongamos 102.400 ficheros de 384 bytes. El tamaño total de los datos es de 37,5 MB. Ahora bien, el disco está lleno, ya que todos los bloques están siendo utilizados! Hay 362,5 MB perdidos. Por lo tanto, es muy importante tener cuidado con el tamaño de bloques, sobre todo si los ficheros que se van a almacenar son de pequeño tamaño.

---

Algunos comandos calculan el tamaño de los ficheros en bloques, como du, df o find. Históricamente, el tamaño de un bloque era de 512 o 1024 bytes. Es una unidad cuyo valor puede cambiar según el comando, y a menudo sin relación con el tamaño de los bloques del sistema de ficheros en el cual usted trabaja. Se impone la prudencia.

---

### b. Superbloque

Cada sistema de ficheros dispone de al menos un superbloque. Un superbloque es una zona de metadatos que contiene varias informaciones en el sistema de ficheros:

- su tipo;
- su tamaño;
- su estado;
- información (posición) en las otras zonas de metadatos (otros superbloques, tabla de inodos, etc.).

Linux intenta primero leer el superbloque primario, el primero del disco. Puede ocurrir que este último esté corrompido después de malas operaciones o una avería. En este caso, ya no se puede acceder a los datos del disco (es imposible saber, por ejemplo, dónde se encuentran los inodos). Un sistema de ficheros de Linux dispone de copias (backups) de los superbloques en varios sitios del disco. Como las escrituras en los diversos superbloques son sincronas, son todos idénticos. Como último recurso, si se suprime uno de ellos, se puede volver a copiar desde otro.

A continuación verá cómo disponer de toda la información en un sistema de ficheros ext2 o ext3.

### c. Tabla de inodos

La palabra **inodo** es la contracción de «índice nodo», o sea nodo de índice. Es una estructura de datos que contiene la información que describe y representa un fichero. Son los **atributos** de un fichero. Cada fichero dispone de un número de inodo (i-number). Todos los inodos están presentes dentro de una tabla de inodos. Se suele repartir partes de esta tabla después de cada superbloque. Una tabla de inodo forma parte de los metadatos.

Un fichero sólo puede tener un único inodo. Un inodo es único dentro de un único sistema de ficheros. Cada

sistema de ficheros dispone de una tabla de inodos independiente. Si el fichero titi lleva el número de inodo 12345 en un primer sistema de ficheros, aunque pepito lleve el número de inodo 12345 en otro, estos ficheros no tienen relación entre ellos.

Sin embargo, veremos más adelante que se puede asignar el mismo número de inodo a dos nombres de ficheros dentro de un mismo sistema de ficheros. Estos dos nombres representan entonces un solo y mismo fichero.

## **Contenido**

El contenido de un inodo cambia de un sistema de ficheros a otro, pero la norma POSIX impone que cada uno de ellos disponga de, al menos, los atributos siguientes para cada fichero:

- su tamaño;
- el identificador del periférico que lo contiene;
- su propietario;
- su grupo;
- su número de inodo;
- su modo (sus derechos) de acceso;
- su fecha de última modificación de inodo (change time);
- su fecha de última modificación de contenido (modification time);
- su fecha de último acceso (access time);
- un contador de hard links (vínculos físicos o duros, ver más adelante).

Un inodo no contiene el nombre del fichero.

Puede obtener información relativa a un inodo con el comando **stat**:

```
# stat capitulo4.doc
  File: `capitulo4.doc'
  Size: 199168          Blocks: 408          IO Block: 4096   fichero regular
Device: 811h/2065d      Inodo: 16629765    Links: 1
Access: (0644/-rw-r--r--) Uid: ( 1000/      seb)  Gid: ( 100/    users)
Access: 2008-03-17 21:39:31.000000000 +0100
Modify: 2008-03-17 21:39:30.000000000 +0100
Change: 2008-03-17 21:40:04.000000000 +0100
```

## **Direcciones**

El inodo contiene también campos de direcciones de bloques repartidos en general entre dos tipos:

- direcciones que apuntan a los primeros bloques de datos del fichero,
- direcciones que apuntan a bloques que contienen otros campos de direcciones,
- y en este último caso de manera recursiva (direcciones de direcciones que apuntan a ellas mismas a otras direcciones), formando un árbol, o bien cada una las hojas (terminaciones) apunta a un bloque de datos. Se habla de bloques **de indirección** (simple, doble, triple).

A continuación presentamos un ejemplo concreto de direcciones en un inodo dentro de un sistema de ficheros ext2.

Un inodo ext2 contiene diez campos que apuntan cada uno a un bloque de datos, y tres campos de

indirección (apuntan a direcciones).

- El primero de estos tres campos apunta, en indirección simple, a 256 direcciones de bloques de datos.
- El segundo apunta, en indirección doble, a 256 direcciones, las cuales apuntan cada una a otras 256 direcciones de bloques de datos, o sea,  $256^2$  bloques de datos.
- El tercer campo apunta, en triple indirección, a 256 direcciones que a su vez apuntan a 256 direcciones, que a su vez apuntan a 256 direcciones de bloques de datos, o sea  $256^3$  bloques de datos.

Si  $n$  es el tamaño de un bloque en bytes, el tamaño máximo de un fichero es, por lo tanto  $n * (10 + 256 + 256^2 + 256^3)$  bytes.

Para un bloque de 4096 bytes, eso representa unos 64 GB!

#### d. Tablas de catálogo

Si un inodo no contiene el nombre del fichero, se coloca éste en una tabla de catálogo. Esta tabla no es más que un directorio. Como un directorio contiene una lista de ficheros y un inodo representa un fichero, entonces cada nombre de fichero se asocia a su inodo dentro del directorio.

Puede representarse esta tabla como un cuadro de dos columnas:

Tabla catálogo dir1 (directorio dir1)	
Inodo	Nombre
12345	Documento.txt
214579	Fichero.doc
47321	Música.mp3
98542	Copia.odt
...	...

#### e. Hard link

Un hard link permite añadir una referencia a un inodo. El hard link añade una asociación en una tabla de catálogo. No se modifican los permisos del fichero.

Un hard link no permite asignar varios nombres a un mismo directorio, ni tampoco permite efectuar vínculos desde o hacia otro sistema de ficheros. Además, tenga cuidado con el contador de vínculos facilitado por el comando **ls -l**: un 1 indica que este fichero no posee otros vínculos o, dicho de otro modo, es el último. Si usted lo suprime, se pierde de manera definitiva. En cambio, mientras este contador sea superior a 1, si se suprime un vínculo, queda una copia del fichero en alguna parte.

```
$ touch fic1
$ ln fic1 fic2
$ ls -li
2394875 -rw-r--r-- 2 seb users    0 mar 21 22:40 fic1
```

```
2394875 -rw-r--r-- 2 seb users 0 mar 21 22:40 fic2
```

ejemplo anterior muestra que los hard links representan el mismo fichero, ya que sólo se trata de los nombres asociados al mismo inodo. Cada uno tiene dos vínculos, lo que es lógico ya que los dos ficheros apuntan al mismo inodo. Finalmente, puede ver que fic1 y fic2 tienen el mismo inodo, a saber 2394875.

## 2. Crear un sistema de ficheros

### a. mkfs, sintaxis general

Los comandos de «formateo» presentes en Microsoft no son similares a los de Linux. Un formato de tipo Microsoft es en realidad la creación y la comprobación de un sistema de ficheros en una partición. La primera etapa consiste en llenar diferentes sectores, bloques y clústers de ceros (o de otro motivo binario) con una verificación del soporte, y la segunda consiste en la escritura de un sistema de ficheros. Esta última operación única basta para la creación de un sistema de ficheros virgen en el disco o la partición.

El comando para crear un sistema de ficheros es **mkfs**. **mkfs** llama a otros programas en función del tipo de sistema de ficheros seleccionado.

```
mkfs -t typefs opciones periférico
```

Es **typefs** el que determina el tipo de sistema de ficheros y, por lo tanto, el programa al que se llama. Existe un programa por tipo de sistema de ficheros:

- **ext2**: mkfs.ext2
- **ext3**: mkfs.ext3
- **ext4**: mkfs.ext4 o mkfs.ext4dev
- **btrfs**: mkfs.btrfs
- **reiserfs**: mkfs.reiserfs
- **vfat**: mkfs.vfat (para todos los formatos FAT, pero existe mkfs.msdos)
- **ntfs**: mkfs.ntfs

 Observe que es muy importante que se indiquen las opciones de cada sistema de ficheros DESPUÉS de haber especificado el sistema de ficheros. Si las especifica antes, serán las opciones de mkfs.

En vez de utilizar mkfs, puede utilizar directamente los programas correspondientes al tipo de sistema de ficheros que se va a escribir.

### b. Un primer ejemplo en ext2

Va a crear un sistema de ficheros de tipo ext2 en la primera partición creada anteriormente, a saber, sdb1. El comando básico es el siguiente:

```
# mkfs -t ext2 /dev/sdb1
mke2fs 1.40.2 (12-Jul-2007)
Etiqueta de sistema de ficheros=
Tipo de sistema operativo: Linux
Tamaño de bloque=4096 (log=2)
Tamaño de fragmento=4096 (log=2)
```

```

125440 i-nodos, 250472 bloques
12523 bloques (5.00%) reservados para el superusuario
Primer bloque de datos=0
Número maximo de bloques del sistema de ficheros=260046848
8 grupos de bloques
32768 bloques por grupo, 32768 fragmentos por grupo
15680 i-nodos por grupo
Superbloques de emergencia almacenados en los bloques:
    32768, 98304, 163840, 229376

```

Escritura de las tablas de i-nodos: completado  
 Escritura de los superbloques y de la información de compatibilidad del sistema de ficheros: completado  
 Se verificará el sistema de ficheros automáticamente cada 35 montajes o después de 180 días, según la primera eventualidad. Utilizar tune2fs -c o -i para sobreescribir el valor.

Esta salida proporciona información interesante:

- Es posible dar una etiqueta (un nombre) al sistema de ficheros.
- Cada bloque es de 4096 bytes.
- Hay 250472 bloques.
- Los nodos i (inodos) representan el número máximo de ficheros: 125440.
- Se reserva el 5% del espacio en disco a root, lo que significa que un usuario lambda no podrá llenar el disco a más del 95%.
- Se reparten las tablas de inodos por grupos.
- Hay un superbloque principal y cuatro superbloques de emergencia (uno por grupo).
- Es posible modificar algunos parámetros del sistema de ficheros con el comando **tune2fs**.

### c. ext2, ext3 y ext4

Como los sistemas de los ficheros ext2 y ext3 son compatibles, comparten los mismos parámetros, entre los cuales los más corrientes son:

Parámetro	Significado
-b	Tamaño de los bloques en bytes, múltiplo de 512. Si no se especifica el tamaño, se determinará por el tamaño de la partición. Cualquier fichero creado en el disco ocupa al menos un bloque y, por lo tanto, si se prevé un gran número de pequeños ficheros, hay que poner un valor bajo (p. ej.: 1024).
-c	Verifique los bloques defectuosos antes de crear el sistema de ficheros. También se puede utilizar el comando <b>badblocks</b> .
-i	Ratio bytes/inodo. Se calcula el tamaño de la tabla de los inodos en función del tamaño total del sistema de ficheros. Un inodo ocupa 128 bytes. El hecho de meter menos limita el número de ficheros posibles, pero permite ganar espacio. -i 4096: un inodo por cada 4 KB.
-m	Porcentaje reservado al superusuario, por defecto el 5%. Ponerlo a cero permite ganar espacio, y root podrá trabajar en ello a pesar de todo.

-L	Label, etiqueta (nombre) del sistema de ficheros, útil para el montaje.
-j	Crea un diario ext3 y por lo tanto un sistema de ficheros ext3.

El ejemplo siguiente crea un sistema de ficheros transaccional ext3 (opción -j) con un tamaño de bloques de 2048 bytes, y un inodo por cada 16 KB. Los usuarios pueden utilizar la totalidad del sistema (no se reserva ningún espacio para root). La etiqueta es DATA.

```
# mkfs -t ext2 -j -b 2048 -i 16384 -m 0 -L "DATA" /dev/sdb1
mke2fs 1.40.2 (12-Jul-2007)
Etiqueta de sistema de ficheros=DATA
Tipo de sistema operativo: Linux
Tamaño de bloque=2048 (log=1)
Tamaño de fragmento=2048 (log=1)
62992 i-nodos, 500944 bloques
0 bloques (0.00%) reservados para el superusuario
Primer bloque de datos=0
Número máximo de bloques del sistema de ficheros=513802240
31 grupos de bloques
16384 bloques por grupo, 16384 fragmentos por grupo
2032 i-nodos por grupo
Superbloques de emergencias almacenados en los bloques:
    16384, 49152, 81920, 114688, 147456, 409600, 442368

Escritura de las tablas de i-nodos: completado
Creación del log (8192 bloques): completado
Escritura de los superbloques y de la información de compatibilidad del
sistema de ficheros: completado
```

Observe que la línea de comandos siguiente tiene el mismo efecto, ya que el sistema de ficheros ext3 induce el parámetro -j:

```
# mkfs -t ext3 -b 2048 -i 16384 -m 0 -L "DATA" /dev/sdb1
```

### **De ext2 a ext3**

Ext3 es un sistema de ficheros ext2 al cual se ha añadido un soporte transaccional. Puede convertir un sistema de ficheros ext2 en ext3 utilizando **tune2fs**.

```
# tune2fs -j /dev/sdb1
tune2fs 1.40.2 (12-Jul-2007)
Creación del i-nodo del diario: completado
Se verificará el sistema de ficheros automáticamente cada 25 montajes
o después de 180 días, según la primera eventualidad. Utilizar tune2fs -c
o -i para sobreescribir el valor.
```

### **De ext3 a ext2**

Para volver a ext2, hay que suprimir otra vez el soporte transaccional con **tune2fs** y el parámetro -O (Option, con O mayúscula):

```
# tune2fs -O ^has_journal /dev/sdb1
```

Verifique la posible presencia de un fichero **.journal** y suprímalo. Finalmente, efectúe una verificación con **fsck**.

## **De ext3 a ext4**

Primero, convierta su sistema de archivos a ext3 añadiendo un diario, como se ha visto anteriormente. Aunque a continuación ya se podría montar directamente este sistema ext3 como ext4, no podría beneficiarse de las ventajas de ext4 si no utilizara los extents. El comando tune2fs permite añadirlos, así como otros parámetros necesarios:

- **extents**: añadir extents.
- **unint\_bg**: optimización de la tabla de inodos que permite reducir las comprobaciones del sistema de ficheros.
- **dir\_index**: modificación de la tabla de catálogos para acelerar el acceso, para directorios de gran tamaño.

```
# tune2fs -O extents,unint_bg,dir_index /dev/sdb1
```

Las optimizaciones no se activarán hasta después de un primer control del sistema de ficheros, lo que verá en la sección Comprobar, ajustar y arreglar. Sin embargo, debe ejecutar el comando siguiente:

```
# fsck -pDf /dev/sdb1
```

Después de esta etapa, el sistema de archivos puede montarse y usarse directamente, como se indica en la sección Acceder a los sistemas de ficheros. Sin embargo, la conversión todavía no es completa, ya que los archivos que están ya en el sistema de ficheros no se han optimizado para usarse con extents. El comando chattr permite modificar los atributos específicos de cada archivo, a los que se les añade los extents con el parámetro +e. Necesitará listar todos los archivos y añadirles este atributo; lo podrá hacer con el comando **find**. Para empezar, monte el sistema de ficheros:

```
# mount /dev/sdb1 /mnt
```

A continuación aplique el atributo +e en cada archivo. El parámetro xdev indica a find que no tiene que descender en subdirectorios que no sean del mismo sistema de ficheros que el inicial:

```
# find /mnt -xdev -type f -exec chattr +e {} \;
```

```
# find /mnt -xdev -type d -exec chattr +e {} \;
```

---

 ¡Atención! Si convierte el sistema de archivos raíz o el que contiene el núcleo de Linux compruebe para empezar si GRUB es compatible con ext4. En caso contrario, corre el riesgo de no poder arrancar su sistema.

---

## **De ext4 a ext3**

Si el sistema de ficheros ext4 no tiene aún la opción de extents, entonces basta con montarlo a ext3. En caso contrario, no se puede realizar la conversión, ya que el atributo, una vez asignado, no debe quitarse. En este caso, la única solución consiste en realizar una copia de seguridad, para poder volver a crear el sistema de ficheros ext3 y restaurar la copia.

### **Label**

Puede visualizar y cambiar la etiqueta del sistema de ficheros tecleando e2label.

```
# e2label /dev/sdb1  
DATA  
# e2label /dev/sdb1 OLDDATA  
# e2label /dev/sdb1
```

Es importante acordarse de modificar las opciones de montaje en consecuencia.

-  Una etiqueta de sistema de ficheros no debe superar los 16 caracteres o será truncada.

## **Reiserfs**

Crear un sistema de ficheros en `reiserfs` es tan simple como con `ext2`, `ext3` o `ext4`. En el ejemplo siguiente, observe que el programa le pide una confirmación antes de crear el sistema de ficheros. A diferencia de lo que se indica, no hay motivo para volver a reiniciar. Hemos suprimido el «blabla» de las primeras líneas.

```
# mkfs -t reiserfs /dev/sdb1
mkfs.reiserfs 3.6.19 (2003 www.namesys.com)
...
Guessing about desired format.. Kernel 2.6.22.17-0.1-default is running.
Format 3.6 with standard journal
Count of blocks on the device: 250464
Number of blocks consumed by mkreiserfs formatting process: 8219
Blocksize: 4096
Hash function used to sort names: "r5"
Journal Size 8193 blocks (first block 18)
Journal Max transaction length 1024
inode generation number: 0
UUID: c1c5ac84-8ce2-4475-829d-f454b7bdb91e
CUIDADO: YOU SHOULD REBOOT AFTER FDISK!
    ALL DATA WILL BE LOST ON '/dev/sdb1'!
Continue (y/n):y
Initializing journal - 0%....20%....40%....60%....80%....100%
Syncing..ok
ReiserFS is successfully created on /dev/sdb1.
```

`reiserfs` acepta parámetros diferentes de `ext2/3`:

Parámetro	Significado
<code>-b</code>	Tamaño de los bloques en bytes, múltiplo de 512 (potencia de 2: 512, 1024, 2048, 4096, 8192) incluido entre 512 y 8192. Si no se especifica el tamaño, se determinará con el tamaño de la partición.
<code>-l</code>	Label, etiqueta (nombre) del sistema de ficheros, útil para el montaje.
<code>-f</code>	Fuerza la ejecución del comando sin hacer preguntas, incluyendo un disco, y no una partición.
<code>-d</code>	Modo debug, facilita más información.

## **Label**

Usted puede modificar una etiqueta `reiserfs` con el comando `reiserfstune`.

```
reiserfstune -l HOME /dev/hda6
```

Acuérdese de modificar las opciones de montaje correspondientes.

## d. xfs

Cree un sistema de ficheros xfs como éste:

```
# mkfs -t xfs -f /dev/sdb1
meta-data=/dev/sdb1          isize=256    agcount=8, agsize=31309
blks
      =
data      =                sectsz=512   attr=0
      =
      =
naming   =version 2        bsize=4096   blocks=250472, imaxpct=25
log       =internal log    bsize=4096   sunit=0 blks, unwritten=1
      =
realtime =none            sectsz=512   sunit=0 blks, lazy-count=0
                                extsz=4096   blocks=0, rtextents=0
```

## e. btrfs

 Atención, btrfs sigue siendo experimental. Es posible que no haya soporte ni herramientas en su distribución, o que estén disponibles mediante la instalación de paquetes adicionales.

La creación de un sistema de archivos btrfs sigue el mismo principio:

```
# mkfs -t btrfs /dev/sdb1
WARNING! - Btrfs Btrfs v0.19 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using
fs created label (null) on /dev/sdb1
nodesize 4096 leafsize 4096 sectorsize 4096 size 401.54GB
Btrfs Btrfs v0.19
```

### De ext3 o ext4 a btrfs

Los comandos btrfs permiten convertir fácilmente un sistema de ficheros ext3 o ext4, con la posibilidad de volver al estado anterior si surgen problemas. En este caso, evidentemente, las modificaciones se borrarán. Para empezar, realice una comprobación del sistema de ficheros:

```
# fsck -f /dev/sdb1
```

continuación ejecute el comando **btrfs-convert**; puede tardar un poco según el tamaño de su sistema de ficheros y la cantidad de datos que contenga:

```
# btrfs-convert /dev/sdb1
creating btrfs metadata.
Creating ext2fs image file.
Cleaning up system chunk.
Conversion complete.
```

Monte el sistema de ficheros:

```
# mount -t btrfs /dev/sdb1 /mnt
```

En este punto de montaje encontrará una imagen del antiguo sistema de archivos en el directorio **ext2\_saved** que se podrá montar si lo desea. Encontrará toda la información necesaria en la wiki de btrfs: [https://btrfs.wiki.kernel.org/index.php/Main\\_Page/#Documentation](https://btrfs.wiki.kernel.org/index.php/Main_Page/#Documentation)

Para liberar el espacio y pasar definitivamente a btrfs sin vuelta atrás, puede borrar el directorio **ext2\_saved**.

## Vuelta atrás

Si btrfs no le convence, y si no ha borrado el directorio **ext2\_saved**, puede anular la conversión, después de haber desmontado el sistema de ficheros:

```
# btrfs-convert -r /dev/sda6  
rollback complete
```

## f. vfat

La creación de un sistema de ficheros VFAT se hace también de la misma manera. Esta vez, usted va a crearlo en la partición sdb5 prevista a este efecto. El comando va a seleccionar automáticamente, en función del tamaño de la partición, el tipo de FAT que se debe crear (12, 16 o 32). Se ha añadido el parámetro **-v** para ver las trazas de su creación.

```
# mkfs -t vfat -v /dev/sdb5  
mkfs.vfat 2.11 (12 Mar 2005)  
Auto-selecting FAT32 for large filesystem  
/dev/sdb5 has 64 heads and 62 sectors per track,  
logical sector size is 512,  
using 0xf8 media descriptor, with 5856706 sectors;  
file system has 2 32-bit FATs and 8 sectors per cluster.  
FAT size is 5709 sectors, and provides 730657 clusters.  
Volume ID is 47df9209, no volume label.
```

 Los comandos **mkfs.vfat** y **mkfs.msdos** son vínculos simbólicos hacia el programa mkdosfs.

Puede especificar varios parámetros, en particular si desea forzar la creación de un tipo de FAT dado:

Parámetro	Significado
-c	Verifica el periférico antes de la creación.
-F	Tamaño de la FAT (12, 16, 32).
-I	Permite utilizar un disco completo, y no una partición (práctico para algunos lectores de MP3).
-n	Nombre del volumen (etiqueta, label).
-v	Visualización de detalles durante la creación.

## Las mtools

Las mtools son herramientas que permiten trabajar en sistemas de ficheros FAT y VFAT como si estuviera bajo MSDOS o la consola de Windows. Retoman la sintaxis de los comandos de origen, pero añaden una **m** delante: mdir, mformat, mlabel, mdeltree, etc.

Se representan los discos y las particiones con letras de unidad **c :**, **d :**, **e :**. Dichas letras pueden representar un disco, una partición o un directorio. Sin embargo, usted debe modificar un fichero de configuración **/etc/mtools.conf**. Por ejemplo, para declarar **/dev/sdb5** como **d:** añada o modifique la línea siguiente:

```
drive d: file="/dev/sdb5"
```

Es útil para modificar después ciertos datos, como el nombre del volumen del sistema de ficheros vfat:

```
# mlabel -s d:  
Volume has no label  
# mlabel d:  
Volume has no label  
Enter the new volume label : DATAFAT  
# mlabel -s d:  
Volume label is DATAFAT
```

# Acceder a los sistemas de ficheros

## 1. mount

El comando **mount** permite acceder a los periféricos de tipo bloque (las particiones) donde se encuentra el sistema de ficheros. El comando **mount** sitúa el sistema de ficheros a montar en un punto del sistema principal llamado punto de montaje (mountpoint).

```
mount -t typefs -o opciones periférico punto_de_montaje
```

### a. Montaje por periférico

Como la partición sdb1 dispone de nuevo de un sistema de ficheros ext3, el comando siguiente monta la raíz del sistema de ficheros que contiene sdb1 en el directorio /mnt/DATA.

```
# mount -t ext3 /dev/sdb1 /mnt/DATA
```

El comando **mount**, cuando se ejecuta sin parámetros, muestra todos los detalles en los sistemas de ficheros actualmente montados (periféricos, sistema de ficheros, punto de montaje, opciones):

```
# mount
/dev/sda6 on / type ext3 (rw,acl,user_xattr)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
debugfs on /sys/kernel/debug type debugfs (rw)
udev on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/sda7 on /home type ext3 (rw,acl,user_xattr)
/dev/sda1 on /windows/C type fuseblk (rw,noexec,nosuid,nodev,noatime,
allow_other,default_permissions,blksize=4096)
securityfs on /sys/kernel/security type securityfs (rw)
/dev/sdb1 on /mnt/DATA type ext3 (rw)
```

Se puede acceder a la misma información visualizando el contenido del fichero **/etc/mtab**.

### Montaje por label

Es tremadamente práctico utilizar etiquetas para los sistemas de ficheros montados en periféricos. Por ejemplo, cuando se procede a reordenar los discos (movimientos en los arrays de discos SCSI, por ejemplo), se modifica la ordenación de los periféricos con respecto al sistema de ficheros principal. Volver a nombrar los periféricos obliga a modificar el fichero **/etc/fstab** para cada uno de los discos. Esto se evita con el uso de etiquetas. Utilice el parámetro **-L** de mount, seguido del nombre del volumen, como en el siguiente ejemplo:

```
# mount -t ext3 -L DATA /mnt/DATA
# mount
...
/dev/sdb1 on /mnt/DATA type ext3 (rw)
```

Se puede obtener la lista de las etiquetas de disco actualmente reconocidas por Linux listando el directorio **/dev/disk/by-label**. Observe que la etiqueta es un vínculo simbólico hacia el fichero periférico correspondiente:

```
# ls -l /dev/disk/by-label/
total 0
```

```
lrwxrwxrwx 1 root root 10 mar 18 14:00 DATA -> ../../sdb1  
lrwxrwxrwx 1 root root 10 mar 18 14:00 DATAFAT -> ../../sdb5
```

### **Montaje por UUID**

Cada sistema de ficheros dispone de un identificador único llamado **UUID**: *Universal Unique Identifier*, en general un número aleatorio codificado en bits lo suficientemente complejo como para que todos sean diferentes en uno o varios sistemas dados. De este modo, si el disco cambia de posición lógica, el UUID no cambia y mount encuentra el sistema de ficheros, cuando en teoría es más posible que dos sistemas de ficheros lleven la misma etiqueta.

Existen varios métodos para conocer el UUID de una partición. Si la herramienta **udev** se utiliza en su Linux, entonces probablemente usted disponga del comando **vol\_id**. Es posible que la ruta hasta el comando no se encuentre en la variable de rutas predefinidas PATH. Éste es el caso de openSUSE, que lo coloca en `/lib/udev`.

```
# ./vol_id -u /dev/sdb1  
67f6e4b8-635c-4103-9a81-877fb7db29fe
```

En las distribuciones más recientes, ya no existe vol-id. El comando **blkid** lo reemplaza eficazmente:

```
# blkid /dev/sdb1  
/dev/sdb1: UUID="67f6e4b8-635c-4103-9a81-877fb7db29fe"  
SEC_TYPE="ext2" TYPE="ext3"
```

Para mostrar sólo el UUID, especifique únicamente este tag con el parámetro `-s UUID` y únicamente el valor con el parámetro `-o value`:

```
# blkid -o value -s UUID /dev/sdb1  
67f6e4b8-635c-4103-9a81-877fb7db29fe
```

Si su sistema de ficheros es de tipo ext2 o ext3, el comando **dumpe2fs** devuelve mucha información, entre la que se encuentra el UUID:

```
# dumpe2fs -h /dev/sdb1 | grep UUID  
dumpe2fs 1.40.2 (12-Jul-2007)  
Filesystem UUID: 67f6e4b8-635c-4103-9a81-877fb7db29fe
```

Finalmente, y al igual que en el caso de las etiquetas, el fichero `/dev/disk/by-uuid` contiene los enlaces simbólicos de los UUID que apuntan al fichero periférico correspondiente:

```
# ls -l /dev/disk/by-uuid/  
total 0  
lrwxrwxrwx 1 root root 10 mar 18 14:00 47DF-9209 -> ../../sdb5  
lrwxrwxrwx 1 root root 10 mar 18 14:00 67f6e4b8-635c-4103-9a81-  
877fb7db29fe -> ../../sdb1  
lrwxrwxrwx 1 root root 10 mar 3 09:23 a1cc2282-b6f4-46e1-bc94-  
91585f1c5872 -> ../../sda5  
lrwxrwxrwx 1 root root 10 mar 3 09:23 C2B072B5B072AF91 -> ../../sda1  
lrwxrwxrwx 1 root root 10 mar 3 09:23 c56b96b5-e52f-453a-ba9d-  
aa1df6f0c3c0 -> ../../sda7  
lrwxrwxrwx 1 root root 10 mar 3 09:23 dd5e92d3-b931-4c18-91a3-  
5edccc57ced9 -> ../../sda6
```

Observe que `vol_id` reconoce los UUID de la mayoría de los sistemas de ficheros, incluyendo FAT y NTFS. Además, tanto `vol_id` como `dumpe2fs` muestran mucha más información. Por ejemplo, `vol_id -l <periférico>` devuelve la label del periférico (volumen name, etiqueta).

---

Para montar un sistema de ficheros por UUID, utilice el parámetro `-U` de mount:

```
# mount -t ext3 -U 67f6e4b8-635c-4103-9a81-877fb7db29fe /mnt/DATA
# mount
/dev/sdb1 on /mnt/DATA type ext3 (rw)
```

### **Volver a montar un sistema de ficheros**

Si necesita modificar alguna de las opciones de montaje, no está obligado a desmontar y volver a montar después el sistema de ficheros. Si modifica una opción de montaje del sistema de ficheros (mediante el parámetro `-o`), puede pasar la opción `remount` para que se tome en cuenta en seguida la modificación. No vuelva a insertar la línea de comandos completa, sino únicamente el periférico o el punto de montaje. En el ejemplo siguiente se vuelve a montar el sistema de ficheros en modo de sólo lectura:

```
# mount -o ro,remount /mnt/DATA
# mount
...
/dev/sdb1 on /mnt/DATA type ext3 (ro)
```

### **b. Opciones de montaje**

Cada sistema de ficheros acepta un cierto número de opciones de montaje que se pueden especificar después del parámetro `-o` de mount. Se separan las opciones por comas. Salvo indicación contraria, las opciones siguientes funcionan con ext2, ext3 y ext4.

Opción	Significado
defaults	Casi siempre presente, la opción <b>defaults</b> sustituye a las opciones <code>rw</code> , <code>suid</code> , <code>dev</code> , <code>exec</code> , <code>auto</code> , <code>nouser</code> y <code>async</code> por defecto.
sync/async	Activa o desactiva las escrituras síncronas. Con <code>async</code> la escritura pasa por una memoria intermedia que la difiere (más eficaz) devolviendo el control más rápido. Es preferible activar la escritura síncrona en soportes externos (claves USB, discos USB/Firewire/eSATA, etc.).
exec/noexec	Permite la ejecución/no ejecución de los ficheros binarios en el soporte.
noatime	Evita la actualización de la marca temporal de acceso a un fichero (aconsejable para los soportes externos, discos SSD, páginas web, newsgroups, etc.).
auto/noauto	Con <code>auto</code> , el sistema de ficheros se monta automáticamente. Con <code>noauto</code> , sólo se puede montar explícitamente (ver fstab).
user/nouser	Cualquier usuario puede montar el sistema de ficheros (implica <code>noexec</code> , <code>nosuid</code> y <code>nodev</code> )/sólo root tiene derecho a montar el sistema de ficheros (ver fstab).
remount	Remontaje del sistema de ficheros para tener en cuenta nuevas opciones.

ro/rw	Montaje en modo de sólo lectura o lectura y escritura.
dev/nodev	Interpretar/No interpretar los ficheros especiales.
noload	Para ext3, no carga el diario.
usrquota/grpquota	Ignorado por el propio sistema de ficheros, pero utilizado por el subsistema de cuotas.
acl	Permite el uso de los Access Control Lists.
user_xattr	Para ext3 y xfs, acepta los atributos extendidos en los ficheros, por ejemplo para pegar en ellos información adicional (la codificación del texto, etc.), campos de indexación (utilizados por Beagle por ejemplo), etc.
umask	Para FAT/NTFS, aplica otra máscara global que la de por defecto (ex 133).
dmask=/fmask=	FAT/NTFS, diferencia las máscaras para los directorios y los ficheros.
uid=/gid=	Como no se gestionan los permisos y propietarios, FAT/NTFS aplica un usuario o un grupo por defecto en los ficheros (ej gid=users).

### c. umount

El comando **umount** desmonta el sistema de ficheros del punto de montaje.

```
# umount /mnt/DATA
```

Si uno o varios archivos del sistema de ficheros por desmontar se están utilizando, **umount** no funcionará. Debe asegurarse de que ningún proceso acceda al sistema de ficheros que pretende desmontar.

```
# umount /mnt/DATA
umount: /mnt/DATA: periférico ocupado
```

El

comando **lsof** le ayuda a determinar qué proceso está utilizando un fichero del punto de montaje en el momento de iniciar el comando. En nuestro ejemplo, el proceso que está accediendo al fichero es el shell bash ejecutado por el usuario seb (probablemente debido a que el directorio actual es /mnt/DATA).

```
# lsof /mnt/DATA
COMMAND PID USER FD TYPE DEVICE SIZE NODE NAME
bash 5366 seb cwd DIR 8,17 4096 2 /mnt/DATA
```

Como administrador, puede de forma violenta iniciar **fuser** para forzar la parada de los procesos que están accediendo al punto de montaje. Es más que probable que no le guste nada al usuario correspondiente (en el caso presentado aquí, se parará su shell y será desconectado).

```
# fuser -km /mnt/DATA
```

### d. /etc/fstab

El fichero `/etc/fstab` contiene una configuración estática de las diferentes opciones de montaje de los sistemas de ficheros. Este fichero es invocado siempre que se inicia el sistema, ya que es aquí donde se especifican los periféricos y sus puntos de montaje. Contiene seis campos.

```
periférico punto_de_montaje typefs opciones dump fsck
```

Los espacios o las tabulaciones hacen de separadores de campos.

Campo	Descripción
periférico	El periférico a montar. Se puede especificar como ruta de periférico ( <code>/dev/hda1</code> por ejemplo), como label de sistema de ficheros si existe ( <code>LABEL=/home</code> ), o incluso como UUID ( <code>UUID=xxxx</code> ).
punto de montaje	El directorio de acceso al sistema de ficheros montado.
typefs	El tipo ( <code>ext2</code> , <code>ext3</code> , <code>ext4</code> , <code>btrfs</code> , <code>reiser</code> , <code>vfat</code> , etc.) del sistema de ficheros.
opciones	Opciones de montaje separadas por comas.
dump	Frecuencia de volcado para las herramientas de copia de seguridad.
fsck	Frecuencia de verificación del sistema de ficheros. 0=ignorar. 1=en primero, 2 en segundo, etc. Se verifican en paralelo los sistemas que tienen el mismo número.

A continuación presentamos un ejemplo (truncado, ya que los sistemas de ficheros virtuales no aparecen) de fichero `/etc/fstab`:

<code>/dev/sda3</code>	<code>/</code>	<code>ext4</code>	<code>acl,user_xattr</code>	<code>1 1</code>
<code>/dev/sda2</code>	<code>/boot</code>	<code>ext4</code>	<code>acl,user_xattr</code>	<code>1 2</code>
<code>/dev/sdb1</code>	<code>/home</code>	<code>ext4</code>	<code>acl,user_xattr</code>	<code>1 2</code>
<code>/dev/sda6</code>	<code>/public</code>	<code>ext4</code>	<code>acl,user_xattr</code>	<code>1 2</code>
<code>/dev/sda1</code>	<code>/windows</code>	<code>ntfs</code>	<code>noauto,users,gid=users,umask=0002,utf8=true</code>	<code>0 0</code>
<code>/dev/sda5</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0 0</code>

En vez de especificar nombres de periféricos estáticos, puede ser preferible especificar una etiqueta (label, volumen) o un UUID.

<code>LABEL=BOOT</code>	<code>/boot</code>	<code>ext4</code>	<code>acl,user_xattr</code>	<code>1 2</code>
<code>UUID=f0bed37c-9ddc-4764-ae7f-133205c36b5d</code>		<code>ext3</code>	<code>acl,user_xattr</code>	<code>1 2</code>

Para terminar, si no desea utilizar ninguna de estas soluciones, pero tiene que usar rutas, puede valerse de los vínculos simbólicos presentes en cada `/dev/disk/by-XXX`, donde XXX representa:

- id: los identificadores de controlador, hardware y partición de los diferentes volúmenes;
- label: los identificadores mediante etiqueta;
- uuid: los identificadores uuid;
- path: los identificadores por ruta física (bus, lun, etc.).

Aquí tiene un ejemplo:

```

./by-id:
lrwxrwxrwx 1 root root 9 mar 20 08:16 ata-HDT722516DLA380_VDKD1CTCE25WSK
-> ../../sda
lrwxrwxrwx 1 root root 10 mar 20 08:16 ata-HDT722516DLA380_VDKD1CTCE25WSK
-part1 -> ../../sda1
lrwxrwxrwx 1 root root 10 mar 20 08:16 ata-HDT722516DLA380_VDKD1CTCE25WSK
-part2 -> ../../sda2
lrwxrwxrwx 1 root root 10 mar 20 08:16 ata-ST3160811AS_6PT1LX2M-part1 ->
../../sdb1
...
./by-label:
lrwxrwxrwx 1 root root 10 mar 20 13:44 EXTERNE -> ../../sdc1
lrwxrwxrwx 1 root root 10 mar 20 08:16 ROOT -> ../../sda3
lrwxrwxrwx 1 root root 10 mar 20 08:16 BOOT -> ../../sda2
...
./by-path:
lrwxrwxrwx 1 root root 9 mar 20 13:44 pci-0000:00:1a.7-usb-0:2:1.0-scsi-
0:0:0 -> ../../sdc
lrwxrwxrwx 1 root root 10 mar 20 13:44 pci-0000:00:1a.7-usb-0:2:1.0-scsi-
0:0:0:0-part1 -> ../../sdc1
lrwxrwxrwx 1 root root 9 mar 20 08:16 pci-0000:00:1f.2-scsi-0:0:0:0 ->
../../sda
...
./by-uuid:
lrwxrwxrwx 1 root root 10 mar 20 08:16 02FCDA46FCDA339F -> ../../sda1
lrwxrwxrwx 1 root root 10 mar 20 13:44 470E-63A6 -> ../../sdc1
lrwxrwxrwx 1 root root 10 mar 20 08:16 527585d3-1e52-4aba-b7fc-70f183884
58d -> ../../sda6
...

```

### **Montaje durante el arranque**

Durante la secuencia de inicio, el fichero `/etc/fstab` es leído por uno de los scripts, casi al principio del boot, entre la carga del núcleo y el inicio de los servicios. Todos aquellos sistemas de ficheros que no tengan configurada la opción `noauto` se montan automáticamente porque el montaje automático está implícito. El primero es el sistema de ficheros raíz `/`. Luego vienen la `swap` y los demás sistemas de ficheros especificados (p. ej.: `/home`, `/usr`, etc.) para terminar con los sistemas virtuales `/proc`, `/sys`, `/dev/pts`, etc.

### **Montaje manual**

Se puede utilizar el contenido de `/etc/fstab` después de la puesta en marcha del sistema para montar y desmontar de manera puntual los sistemas de ficheros que no disponen, por ejemplo, de la opción `noauto`, o los soportes masivos como los lectores CD/DVD. En este caso, use el comando `mount` pasándole simplemente la etiqueta, los puntos de montaje o el periférico sin tener que volver a escribir toda la línea de comandos. `Mount` buscará el resto de las opciones en `/etc/fstab`.

```

mount /home
mount -L /u01
mount LABEL=/boot
mount /dev/hda5

```

### **Montarlo todo**

Si ha realizado modificaciones importantes en la `fstab`, como el añadido de varios puntos de montaje, puede, en vez de montar cada sistema de ficheros uno por uno, montarlos todos con el parámetro `-a` de `mount`:

```
# mount -a
```

### **bind**

La opción particular bind puede ser muy práctica para montar una parte de sistema de ficheros en varios puntos de montaje. Permite evitar el uso de vínculos simbólicos y sus defectos (modificación del fichero al que se señala y nombre del propio vínculo). En el ejemplo siguiente se relaciona el sistema de ficheros que tiene el label **u01** con el directorio **/u01**. Luego se relaciona **/u01/applis** con el directorio **/applis**.

```
LABEL=/u01      /u01          ext3    defaults 1 2  
/u01/applis   /applis       none    bind
```

## e. El caso de los CD e imágenes ISO

Los CD-Rom, DVD-Rom y otros soportes de este tipo se montan como cualquier otro soporte de bloques. Los CD-Rom y algunos DVD-Rom utilizan el sistema de ficheros **iso9660**.

```
# mount -t iso9660 /dev/sr0 /media/cdrom
```

La mayoría de los DVD-Rom utilizan, sin embargo, el formato **UDF** (*Universal Disk Format*).

```
# mount -t udf /dev/sr1 /media/dvd
```



Las distribuciones Linux recientes se liberan del montaje manual de los soportes externos, ya que se trate de un CD, DVD, clave USB o disco externo. Los servicios udev se encargan en el momento de la conexión o de la inserción del soporte de crear los ficheros especiales asociados, y luego de montar y desmontar los soportes de manera automática.

Un fichero ISO es una imagen del contenido de un CD o DVD. Es un sistema de ficheros iso9660 o udf en un fichero. Es posible utilizar esta imagen como periférico con ayuda de la opción loop. Esta técnica consiste en relacionar la imagen con un periférico en modo loopback, y conseguir así que las herramientas lo traten como un disco más.

```
# mount -o loop -t iso9660 image.iso /mnt/iso
```

# Controlar el sistema de ficheros

## 1. Estadísticas de ocupación

### a. Por sistema de ficheros

El comando **df** permite obtener estadísticas de ocupación de cada sistema de ficheros montado. Sin argumento, **df** facilita información sobre todos los sistemas de ficheros. Puede pasar como argumento un periférico montado o un punto de montaje. Si pasa un directorio cualquiera, df da la información del sistema de ficheros que contenga este directorio.

```
# df
Sis. de fich. 1K-bloques Ocupado Disponible Capacidad Montado en
/dev/sda3      41286828  6482952  32706592  17%      /
udev          1031240   124     1031116   1%      /dev
/dev/sda2      521780   27092   468184    6%      /boot
/dev/sdb1      153834852 49189572  96830864  34%      /home
/dev/sda6      73142560  19150372  50276760  28%      /public
/dev/sdc1      292890560 175894672 116995888  61%      /media/EXTERNO
```

El resultado es explícito. La unidad por defecto es el KB (idéntico al parámetro **-k**) aunque la norma POSIX define una unidad de bloque en 512 bytes. Puede modificar los parámetros para pedir el resultado en MB (**-m**).

```
# df -m /home
Sis. de fich. 1M-bloques Ocupado Disponible Capacidad Montado en
/dev/sdb1      150230   48043   94557    34%      /home
```

Para que sea más legible, añada el parámetro **-h** (*Human readable*).

```
# df -h /home
Sis. de fich. Tam. 0c. Disp. %0c. Montado en
/dev/sdb1      147G 47G  93G  34%      /home
```

No confunda este último parámetro con **-H**, que visualiza el resultado en unidades **SI** (*Sistema Internacional*).

```
# df -H /home
Sis. de fich. Tam. 0c. Disp. %0c. Montado en
/dev/sdb1      158G 51G  100G 34%      /home
```

 Las unidades del SI que definen las unidades de peso y medida se basan en potencias de 10. Es sencillo memorizar que 1 Kg es igual a  $10^3$  gramos, o sea, 1000 gramos. Por lo tanto, 1 KB vale  $10^3$  bytes, o sea 1000 bytes... ¿No está de acuerdo? Un ordenador no trabaja con potencias de 10, sino de 2. A nivel físico, 1 KB equivale  $2^{10}$  bytes, o sea 1024 bytes; 1 MB vale  $2^{20}$  bytes, y así sucesivamente. Este método se llama método tradicional. El sistema internacional prefiere emplear los términos kibibyte (kilo Binario, Kib), mebibbytes (Meb) y gibibbytes (Gib) para las representaciones binarias, y KB, MB y GB para las potencias de 10, como en el caso de los metros y los gramos. Ahora entiende por qué un disco de 160 GB se corresponde en realidad a 152,5 Gib. En cierto modo, nos dejamos engañar de manera legal y oficial.

La **-T** añade la visualización del tipo de sistema de ficheros.

```
# df -T /home
Sis. de fich. Tipo 1K-bloques Ocupado Disponible Capacidad Montado en
/dev/sdb1 ext3 153834852 49197688 96822748 34% /home
```

El

comando **df** permite también facilitar estadísticas para el uso de los inodos. Puede combinar los parámetros **-i** y **-h**.

```
# df -i /home
Sis. de fich. Inodos IUtil. ILib. %IUtil. Montado en
/dev/sdb1 19546112 86016 19460096 1% /home
# df -ih /home
Sis. de fich. Inodos IUtil. ILib. %IUtil. Montado en
/dev/sdb1 19M 84K 19M 1% /home
```

## b. Por estructura

El comando **du** (*disk usage*) facilita la información relativa al espacio ocupado por una estructura (un directorio y todo su contenido). Si no se especifica nada, se utiliza el directorio corriente. Los parámetros **-k** (Kb) y **-m** (MB) determinan la unidad. Se facilita el tamaño para cada elemento (incluso redondeado). El tamaño total de la estructura está en la última línea.

```
# du -m LIBRO_ALGO
1 LIBRO_ALGO/BACKUP/capítulo7/codigo_java
2 LIBRO_ALGO/BACKUP/capítulo7
1 LIBRO_ALGO/BACKUP/Introducción
1 LIBRO_ALGO/BACKUP/capítulo4/ilustraciones
...
42 LIBRO_ALGO/
```

Para obtener el total, y no todos los detalles, utilice **-s**.

```
# du -ks LIBRO_ALGO
42696 LIBRO_ALGO/
```

Observe que no se limita **du** a un único sistema de ficheros y sigue calculando si encuentra un punto de montaje en la estructura que analiza. Si quiere limitar el cálculo al sistema de ficheros corriente sin entrar en los puntos de montaje presentes en la estructura, especifique **-x**.

```
# du -msx /
1064 /
```

## 2. Comprobar, ajustar y arreglar

### a. fsck

El comando **fsck** permite comprobar y arreglar un sistema de ficheros.

**fsck -t typefs periférico**

El sistema de ficheros que se quiere comprobar o arreglar no debería estar montado, o, como mucho, montado en modo de sólo lectura.

De la misma forma que **mkfs**, **fsck** invoca a otro comando teniendo en cuenta el tipo del sistema de ficheros para comprobar: esos otros comandos más especializados son **fsck.ext2**, **fsck.ext3**, etc. Cada uno puede presentar opciones particulares. Si **fsck** no reconoce la opción que se le proporciona, la transmite al programa correspondiente. Si usted no indica un tipo, **fsck** intenta determinarlo por sí mismo.

Para este ejemplo, se pasa el parámetro **-f** a **fsck** para forzar la comprobación (no ha sido posible producir una corrupción), así como el parámetro **-v** para facilitar todos los detalles.

```
# fsck -fV /dev/sda2
fsck 1.40.2 (12-Jul-2007)
e2fsck 1.40.2 (12-Jul-2007)
Paso 1: verificación de los i-nodos, de los bloques y de los tamaños
Paso 2: verificación de la estructura de los directorios
Paso 3: verificación de la conectividad de los directorios
Paso 4: verificación de los contadores de referencia
Paso 5: verificación de la información del sumario del grupo

42 inodes used (0.06%)
  1 non-contiguous inode (2.4%)
    # of inodes with ind/dind/tind blocks: 10/1/0
8864 blocks used (6.69%)
  0 bad blocks
  1 large file

27 regular files
  3 directories
  0 character device files
  0 block device files
  0 fifos
  0 links
  3 symbolic links (3 fast symbolic links)
  0 sockets
-----
33 files
```

Cuando el sistema de ficheros está dañado, **fsck** inicia una batería de preguntas por cada acción necesaria. Puede pasar el parámetro **-p** para intentar una reparación automática, o también **-y** para forzar las respuestas a sí.

Durante el inicio del sistema, éste comprueba desde hace cuánto tiempo, o después de cuántos montajes, no se ha comprobado el sistema de ficheros. Si el intervalo de tiempo es demasiado importante, entonces ejecutará un **fsck** en el sistema de ficheros correspondientes. Se pueden modificar los intervalos mediante el comando **tune2fs**.

## b. badblocks

El comando **badblocks** intenta comprobar los bloques defectuosos en el periférico de almacenamiento proporcionado como argumento. **mkfs** o **fsck** pueden llamar a este comando si se les proporciona el parámetro **-c** (check).

Por defecto, **badblocks** lee la totalidad de los bloques del soporte y devuelve un error si uno o varios de ellos son ilegibles. Se puede ejecutar el comando incluso aunque el sistema de ficheros esté montado, excepto si usted intenta una prueba en lectura y escritura, incluso no destructiva.

```
# badblocks -v /dev/sda2
Comprobación de los bloques 0 a 530144
Comprobación de los bloques defectuosos (prueba en modo de sólo lectura): done
Paso completado, 0 bloques defectuosos localizados.
```

Los parámetros **-n** (no destructivo) y **-w** (write, con motivos, destructivo) intentan escribir en los bloques. El primero lee y vuelve a escribir lo leído en el bloque, el segundo escribe otro tipo de información (0xaa, 0x55, 0xff, 0x00) y por lo tanto sobreescribe lo anterior.

La ejecución de **badblocks** puede ser larga, varias horas en algunos centenares de GB.

### c. dumpe2fs

El comando **dumpe2fs** acepta como argumento un periférico que contiene un sistema de ficheros ext2, ext3 o ext4. Devuelve un gran número de información sobre el sistema de ficheros.

```
# dumpe2fs /dev/sda2
dumpe2fs 1.40.2 (12-Jul-2007)
Filesystem volume name: /boot
Last mounted on: <not available>
Filesystem UUID: abc32a5a-a128-4492-8e03-248521015835
Filesystem magic number: 0xEF53
Filesystem revision #: 1 (dynamic)
Filesystem features: has_journal resize_inode dir_index filetype
needs_recovery sparse_super large_file
Filesystem flags: signed directory hash
Default mount options: (none)
Filesystem state: clean
Errors behavior: Continue
Filesystem OS type: Linux
Inode count: 66400
Block count: 132536
Reserved block count: 6626
Free blocks: 123672
Free inodes: 66358
First block: 0
Block size: 4096
Fragment size: 4096
Reserved GDT blocks: 32
Blocks per group: 32768
Fragments per group: 32768
Inodes per group: 13280
Inode blocks per group: 415
Filesystem created: Sat Feb 23 22:52:05 2008
Last mount time: Thu Mar 20 19:13:51 2008
Last write time: Thu Mar 20 19:13:51 2008
Mount count: 1
Maximum mount count: 500
Last checked: Thu Mar 20 19:07:48 2008
Check interval: 5184000 (2 months)
Next check after: Mon May 19 20:07:48 2008
Reserved blocks uid: 0 (user root)
Reserved blocks gid: 0 (group root)
First inode: 11
Inode size: 128
Journal inode: 8
Default directory hash: tea
Directory Hash Seed: f1584155-5760-4445-8009-0444ffa81f91
Journal backup: inode blocks
Tamaño del journal: 16M

Grupo 0: (Bloques 0-32767)
superbloque Primario a 0, Descriptores de grupos a 1-1
Bloques reservados GDT a 2-33
Bitmap de bloques a 34 (+34), Bitmap de i-nodos a 35 (+35)
Tabla de i-nodos a 36-450 (+36)
25696 bloques libres, 13256 i-nodos libres, 2 directorios
Bloques libres: 4559-5660, 6767-12287, 12289-14335, 15644-26721, 26820-32767
I-nodos libres: 14, 16-20, 23-24, 33-13280
...
```

Esta salida es muy larga (y ello a pesar de que ha sido truncada), pero le da todos los detalles posibles en el sistema de ficheros. Puede aislar únicamente el encabezamiento (hasta el tamaño del diario) con el parámetro **-h**. Si busca una información específica, lo mejor es utilizar el comando **grep**.

```
# dumpe2fs -h /dev/sda2|grep -i "block size"
dumpe2fs 1.40.2 (12-Jul-2007)
Block size:          4096
```

#### d. tune2fs

El comando **tune2fs** permite modificar algunos parámetros de un sistema de ficheros ext2 o ext3. Hemos visto anteriormente este comando cuando se explicó cómo convertir ext2 a ext3, y viceversa. Le presentamos algunos parámetros soportados por el comando:

Parámetro	Significado
-c n	Número de veces que se debe montar el sistema de ficheros antes de ser comprobado automáticamente. Por ejemplo, si n vale 10, se iniciará de manera automática fsck la décima vez que se monte. Si n vale 0 o -1, se desactivará la verificación.
-i n	Intervalo de tiempo entre dos comprobaciones. La unidad por defecto es el día. Se pueden juntar los sufijos d (días), w (semanas) o m (mes) al número. -i 180d significa que se controlará el sistema de ficheros después de 180 días.
-j	Añade un diario sobre un sistema de ficheros ext2. Es preferible hacerlo con el sistema desmontado. En el caso contrario, se añade un fichero oculto <code>.journal</code> a la raíz del sistema, inmutable (salvo si usted destruye el diario), que se integrará dentro del sistema durante la próxima ejecución de fsck.
-L	Modifica la etiqueta (label, nombre de volumen). La etiqueta no debe superar 16 caracteres o será truncada.
-e err	Indica cómo debe reaccionar el núcleo si se detecta un error en el sistema de ficheros durante el boot. El valor por defecto es «continuo». Los otros valores posibles son «panic» (bloqueo del núcleo en modo kernel panic) y «remount-ro»: remontaje en modo de sólo lectura.
-m n	El porcentaje n representa el tamaño reservado a los procesos iniciados por root (y el propio root) sobre el tamaño total de la partición. En un sistema de ficheros reservados a los usuarios, poner 0 permite llenar el sistema de ficheros hasta 100%. Pero es importante reservar en la raíz, o /var, una zona para que algunas bitácoras como syslogd puedan seguir escribiendo los registros. Por defecto se reserva el 5%.
-o [^]option	Añade o suprime (con ^) la opción indicada por defecto al montaje. Las opciones pueden ser, por ejemplo, acl o user_xattr.
-O [^]function	Añade o suprime (con ^) la función indicada. La función más famosa es «has_journal». -O has_journal equivale a -j. -O ^has_journal convierte ext3 en ext2.
-U UUID	Modifica el valor del UUID a su conveniencia (formato hexadecimal). Es posible suprimirlo (clear), generar uno de manera aleatoria (random) o generar uno en función de la fecha (time).

-s 0/1	Activa o desactiva la «sparse super feature». En discos de gran tamaño, la activación reduce el número de bloques de emergencia para ganar espacio. Después debe ejecutar fsck.
--------	---

```
# tune2fs -m 0 -s 1 -U random -e remount-ro -c 60 -i 180 /dev/sdb1
```

---

- Observe que **-c** e **-i** van de la mano. A vencimiento del plazo se efectúa la comprobación con fsck. Luego el sistema pone de nuevo a cero las fechas y los contadores. Como consecuencia de un fsck, el sistema pone de nuevo a cero las fechas y los contadores. Se verifican los contadores durante el montaje en el momento del inicio (boot) del sistema. Si hace 300 días que el sistema no se ha reiniciado y no se ha verificado el sistema de ficheros durante este intervalo, no se verificará automáticamente durante estos 300 días, sino en el próximo reinicio. Como algunos servidores reinician pocas veces (por ejemplo, cada dos años), no se comprueba el sistema de ficheros de manera automática durante todo este tiempo...
-

# La swap

## 1. ¿Porqué crear una swap?

En un entorno de 32 bits, un proceso puede acceder teóricamente a 4 GB de espacio de memoria. Dispone de 4 GB de memoria virtual para él de forma exclusiva. Hay varios límites a esta posibilidad:

- El espacio de memoria direccionable de un proceso se comparte entre la zona de código y la zona de datos, cuyo tamaño puede variar según el núcleo utilizado.
- No todos los ordenadores disponen de 4 GB de memoria (aunque sea habitual encontrar servidores de Linux que disponen de 16, 32 o incluso 64 GB de memoria).
- Todos los procesos deben compartir la memoria del ordenador.

En un sistema de 64 bits no hay límite de 4 GB, pero el tamaño de la memoria física disponible sigue siendo fijo.

¿Qué ocurre si un proceso no tiene bastante memoria para tratar sus datos? El sistema operativo descargará segmentos de la memoria física en una zona de intercambio en disco que hará de memoria virtual intermedia. Por lo tanto, hay un intercambio entre la memoria física y esta zona de intercambio, llamada espacio swap. Este proceso permite utilizar más memoria de la que dispone realmente el ordenador, a costa de una importante ralentización del programa si éste resulta ser muy «devorador».

## 2. Tamaño óptimo

No hay reglas estrictas para establecer el tamaño de la swap. Sin embargo las reglas corrientes siguientes son válidas en la mayoría de los casos:

- Si la RAM tiene menos de 512 MB, la swap debe ser el doble de grande.
- Si la RAM tiene entre 1 y 4 GB, la swap debe tener el tamaño de la RAM.
- Si la RAM supera las 4 GB, la swap debe tener 4 GB más o menos, según el uso de los procesos.

Algunos servidores requieren bastante más espacio, como por ejemplo swaps de 8 o 16 GB, o incluso más. En este caso, conviene recurrir a otros modelos de arquitectura hardware o software.

## 3. Crear una partición de swap

- Ya sabe crear una partición. Cree una partición con fdisk del tamaño deseado para la swap y asígnele el tipo **82**.
- Sincronice la tabla de las particiones con **partprobe**.
- Utilice el comando **mkswap** para preparar a la partición que se va a recibir de la swap.

```
# mkswap /dev/sda5
Inicialización de la versión del espacio de swap 1, tamaño = 2154983 kB
ninguna etiqueta, UUID=c84714e6-c42c-44d4-9fe7-10dc6afac644
```

- Su swap está lista.

Es posible asignar una etiqueta a la partición de swap con el parámetro **-L**.

 Si crea más de 1 o 2 GB de swap, debería pensar en crear varias particiones de swap en discos diferentes ubicados en varios controladores físicos. Linux utilizará cada una de estas particiones, lo que asegura accesos más rápidos.

## 4. Activar y desactivar la swap

### a. Activación dinámica

Linux permite activar y desactivar la swap, o partes de la swap, directamente sin tener que volver a iniciar el sistema.

El comando **swapon** permite activar una partición de swap:

```
# swapon /dev/sda5
```

El parámetro **-p** permite modificar la prioridad de la swap. Cuanto más elevado sea el valor, entre 0 y 32767, más alta será la prioridad de una zona de swap. Linux usa un sistema de prioridades. Este parámetro es útil cuando maneja varias particiones de swap en diferentes discos. En tal caso, dé prioridad al disco más rápido. Si las particiones de swap tienen asignadas las mismas prioridades, Linux repartirá la carga por igual.

Como con mount, el parámetro **-L** permite activar una zona de swap gracias a su etiqueta.

El comando **swapoff** desactiva una zona de swap. Tenga cuidado en disponer del espacio de memoria libre necesario; si no, el comando no funcionará.

El contenido de **/proc/swaps** refleja el estado actual de las zonas de swap activas.

```
# cat /proc/swaps
Filename           Type      Size    Used   Priority
/dev/sda5          partition 1461872 2012      -1
```

### b. En /etc/fstab

Las zonas de swap se colocan en el fichero **/etc/fstab**. Aquí tiene un ejemplo:

```
/dev/sda5 swap swap defaults 0 0
```

Se pueden especificar las opciones **noauto** y **pri=X**. La opción **pri** permite definir la prioridad de la zona de swap.

En el momento de iniciarse, el sistema ejecuta **swapon -a**, que activa todas las particiones de swap presentes en la fstab, excepto si se especifica noauto.

En el momento de la parada, el sistema ejecuta **swapoff -a**, que desactiva completamente la swap.

## 5. En caso de emergencia: fichero de swap

¿Qué hacer si le falta espacio de swap y ya no es posible crear una nueva partición? Linux sabe utilizar un fichero de intercambio (como Windows, incluso el de Windows). Si queda espacio en uno de sus sistemas de ficheros, puede crear encima un fichero de intercambio con un tamaño predefinido. Esta swap será menos eficaz que una partición de swap (problema de fragmentación, tiempo de acceso al sistema de ficheros).

Aquí tenemos las operaciones para una pequeña swap de 32 MB:

```

# free | grep Swap
Swap: 2104472      4344    2100128
# dd if=/dev/zero of=/swap bs=1024 count=32768
32768+0 grabaciones leídas
32768+0 grabaciones escritas
33554432 bytes (34 MB) copied, 0,35697 s, 94,0 MB/s
slyserver:~ # mkswap /swap
Inicialización de la versión del espacio de swap 1, tamaño = 33550 kB
ninguna etiqueta, UUID=b2e5e99e-09a1-4b2d-ac76-59f76526453a
slyserver:~ # chmod 600 /swap
slyserver:~ # sync
slyserver:~ # swapon -v /swap
swapon sur /swap
slyserver:~ # free | grep Swap
Swap: 2137232      4308    2132924

```

Modifique si es preciso el fichero **/etc/fstab** para activar esta swap después del montaje de los sistemas de ficheros. Se activa la swap en el boot, en general después del montaje de /. Si la nueva swap se encuentra en otra parte (otro punto de montaje), obtendrá un mensaje de error porque la swap ha sido activada antes que los otros puntos de montaje. Por lo tanto, es preferible crear el fichero en el sistema de ficheros raíz /.

```
/swap swap swap defaults 0 0
```

## 6. Estado de la memoria

### a. free

El comando **free** le proporciona la información relativa a la memoria física (RAM) de su ordenador, así como a la ocupación de la swap. La unidad por defecto es el KB (idéntico con el parámetro **-k**), pero se puede modificar en MB (**-m**) e incluso en GB (**-g**).

	total	used	free	shared	buffers	cached
Mem:	2062484	2045732	16752	0	707512	776528
-/+ buffers/cache:		561692	1500792			
Swap:	2104472	132	2104340			

Sin embargo, tenga cuidado en interpretar correctamente las columnas. En el resultado anterior, aun disponiendo de 2 GB de memoria, el sistema indica que sólo hay libres 16 MB. Esto se debe a que Linux tiende a reservarse todo el espacio disponible en forma de buffers (memoria intermedia) y de caché. El contenido de la caché es volátil; por lo tanto Linux puede liberar en gran medida este espacio para asignarlo a los programas y datos. También es el caso para los buffers. Cuando se insertó el comando anterior, el ordenador estaba copiando una pista de DVD en el disco duro. Unos instantes después de finalizar la copia, el resultado es el siguiente:

	total	used	free	shared	buffers	cached
Mem:	2062484	1586772	475712	0	9708	996000
-/+ buffers/cache:		581064	1481420			
Swap:	2104472	44	2104428			

Se han liberado los buffers, una parte ha sido devuelta a la caché justo al principio de la codificación de la pista en DivX. Por lo tanto, durante sus cálculos de memoria libre, tenga en cuenta el hecho de que se pueden liberar las cachés. El sistema dispone de unos 1,4 GB de espacio de memoria (sumando la libre y la cacheada) que Linux puede asignar a los programas.

El parámetro **-m** muestra el resultado en MB, mientras que **-g** lo muestra en GB. El parámetro **-t** muestra el

total por cada columna.

### **Highmem, lowmem**

Si trabaja en un entorno de 32 bits, un parámetro muy interesante es **-l**, que muestra el detalle de la ocupación de memoria de zonas de memoria alta (highmem) y baja (lowmem). En estos sistemas, la memoria se descompone en las zonas baja y alta. La zona highmem es la zona en la que los procesos tienen su código y sus datos. La zona lowmem es una zona reservada al núcleo y que sirve, entre otras funciones, para almacenar diversas cachés y buffers, así como las tablas de asignación de memoria, para cada proceso. Esta memoria puede también asignarse a los procesos del núcleo. Si un proceso reserva memoria, los datos relativos a su posición y su tamaño reales están en lowmem. De igual modo, todas las transferencias de datos (entrada y salida de disco, por ejemplo) sólo las puede hacer el núcleo y, por lo tanto, estos datos deben transitar de la zona highmem a la zona lowmem.

Esta diferenciación puede conducir a casos sorprendentes donde el sistema operativo puede necesitar memoria y desencadenar mecanismos de autodefensa (matando procesos) mientras que la memoria no parece saturada y el swap no está en uso. El comando free muestra por defecto la suma de ambas zonas de memoria. Lo que sucede realmente es que la zona lowmem está llena. El siguiente capítulo le mostrará cómo reducir los riesgos de que esto se produzca.

```
$ free -l
      total        used        free      shared  buffers     cached
Mem:   1025488      735324      290164          0       32868    419224
Low:  886408  596528  289880
High: 139080 138796    284
-/+ buffers/cache: 283232    742256
Swap: 1999868        0  1999868
```

Los sistemas en 64 bits no tienen este problema, ya que pueden acceder a toda la memoria, que es de tipo lowmem.

### **b. Memoria reservada**

2 GB corresponden a 2097152 KB. Ahora bien: el total presenta una diferencia de unos 34 MB. Se reserva esta memoria para el núcleo del sistema y el resto de los binarios no la pueden utilizar. Es útil para los tratamientos del núcleo, su carga, el initrd. Vea el resultado del comando siguiente (truncado de manera voluntaria):

```
# dmesg | grep -i memory
...Memory: 2057756k/2096640k available (2053k kernel code, 38496k
reserved, 1017k data, 316k init)
Freeing initrd memory: 4411k freed
Freeing unused kernel memory: 316k freed
...
```

sistema se reserva unos 38 MB de memoria, luego libera la memoria que ya no necesita para obtener el resultado esperado.

### **c. meminfo**

El sistema de ficheros virtual **/proc** contiene información detallada sobre la memoria mediante el seudofichero **/proc/meminfo**. Parece bastante difícil encontrar algo más completo. La salida siguiente es el resultado en un sistema Linux de 64 bits. En 32 bits, dos líneas adicionales (highmem y lowmem) indican las zonas reservadas a los datos y al núcleo. Las primeras líneas y las relativas a la swap son idénticas al resultado del comando **free**.

```
# cat /proc/meminfo
MemTotal:      2062484 KB
MemFree:       16452 KB
Buffers:        4152 KB
Cached:        1456476 KB
SwapCached:     0 KB
Active:        1307912 KB
Inactive:      614356 KB
SwapTotal:     2104472 KB
SwapFree:      2104428 KB
Dirty:         188324 KB
Writeback:      0 KB
AnonPages:     461688 KB
Mapped:        120268 KB
Slab:          62476 KB
SReclaimable:  38384 KB
SUnreclaim:    24092 KB
PageTables:    15344 KB
NFS_Unstable:   0 KB
Bounce:         0 KB
CommitLimit:   3135712 KB
Committed_AS:  845496 KB
VmallocTotal:  34359738367 KB
VmallocUsed:   56016 KB
VmallocChunk:  34359678971 KB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
Hugepagesize:   2048 KB
```

# Requisitos y objetivos

## 1. Requisitos

- Disponer de la contraseña del root.
- Saber operar con el shell.
- Disponer de un editor de texto.
- Disponer de uno o varios periféricos hotplug.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Entender el proceso de boot.
- Efectuar los ajustes básicos de la BIOS.
- Configurar el gestor de arranque GRUB.
- Entender la carga del núcleo y del initrd.
- Controlar el funcionamiento de init y del inittab.
- Gestionar los servicios y runlevels.
- Consultar los registros del sistema.
- Gestionar el núcleo y sus módulos.
- Modificar los parámetros dinámicos del núcleo.
- Compilar e instalar un nuevo núcleo.
- Entender los ficheros periféricos.
- Ver los recursos físicos de su máquina.
- Entender el soporte del USB y del Hotplug.
- Conocer el principio de las reglas udev.

# Prácticas

## 1. GRUB y el proceso de boot

Objetivo: iniciar Linux en un modo mínimo en caso de pérdida de contraseña. Esta «artimaña» no parece funcionar en Ubuntu.

1. Encienda su PC y deténgase en el menú de GRUB. Si está en modo gráfico, pulse [Esc]. En la entrada por defecto, pulse la tecla e para editar la información de boot de este menú.
2. Con las teclas de dirección, vaya a la línea que empieza por «kernel» y pulse la tecla e para editarla.
3. Al final de esta línea, añada init=/bin/sh y pulse [Intro]. Finalmente, pulse b para bootejar.
4. Despues de unos segundos aparece un prompt. Pase al teclado español (qwerty por defecto).

```
# loadkeys es
```

5. Vuelva a montar el sistema de ficheros raíz en sólo lectura:

```
# mount -o remount,rw /
```

6. Teclee ahora «passwd» y valide: puede cambiar la contraseña de root.

```
# passwd
```

## 2. init y runlevel

Objetivo: manejar la configuración de init y de los niveles de ejecución. En sistemas Red Hat y asimilados (Mandriva, openSUSE, Fedora) y GRUB.

1. Si arranca en modo gráfico, modifique /etc/inittab para iniciar en el nivel 3. Cambie el valor de 5 a 3 en la línea initdefault y vuelva a bootejar.

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

2. Liste los servicios que se inician en el nivel 3:

```
# ls -l /etc/init.d/rc3.d/S*
```

3. Si el servicio sshd no arranca, actívelo en los niveles 3 y 5:

```
# chkconfig -list sshd  
# chkconfig sshd --level 35
```

4. chkconfig no inicia el servicio, pero lo activa para el próximo boot o cambio de nivel. Inicie sshd:

```
# service sshd start
```

5. Pase a la consola virtual 6 con [Alt][F6] (o [Ctrl][Alt][F6] bajo X). Conéctese y luego desconéctese. ¿Por qué el terminal vuelve a su punto de partida? Porque /etc/inittab contiene

el comando respawn, que permite al proceso iniciarse de nuevo si había finalizado.

6. Con el comando shutdown, apague el ordenador ahora. ¿Cuál es el nivel de ejecución activado?

```
# shutdown -r now
```

Es el nivel 0, de parada, el que está activado.

### 3. Núcleo y módulos

Objetivo: gestionar los parámetros dinámicos del núcleo y los módulos.

1. Verifique su versión del núcleo de Linux y diríjase al directorio de sus módulos:

```
# uname -r  
2.6.25.5-1.1-default  
# cd /lib/modules/2.6.25.5-1.1-default
```

2. Verifique la fecha del fichero módulos.dep. Si parece antigua, inicie un comando para regenerarla.

```
# ls -l modules.dep  
-rw-r--r-- 1 root root 385503 junio 18 20:41 modules.dep  
# depmod -a  
# ls -l modules.dep  
-rw-r--r-- 1 root root 385601 junio 29 20:12 modules.dep
```

3. Liste los módulos actualmente cargados. Si no está el módulo vfat, cárguelo con sus dependencias.

```
# lsmod  
# lsmod | grep vfat  
# modprobe vfat
```

4. De la misma manera, descargue el módulo vfat y sus dependencias.

```
# modprobe -r vfat
```

5. El parámetro dinámico arp\_announce del núcleo permite modificar los encabezamientos ARP en función de la dirección de destino del paquete. En una máquina que dispone de varias tarjetas de red, el valor por defecto puede ocasionar problemas, ya que Linux puede contestar con la dirección de una tarjeta cualquiera. La tarjeta tiene que contestar con una dirección de la misma subred que la de destino. Verifique qué parámetros del núcleo se ven involucrados:

```
# sysctl -a | grep arp_announce  
net.ipv4.conf.all.arp_announce = 0  
net.ipv4.conf.default.arp_announce = 0  
net.ipv4.conf.lo.arp_announce = 0  
net.ipv4.conf.eth0.arp_announce = 0  
net.ipv4.conf.eth1.arp_announce = 0  
net.ipv4.conf.eth2.arp_announce = 0
```

6. Modifique de manera dinámica para el conjunto de los adaptadores el valor arp\_announce a 1. El manual de sysctl indica que se debe especificar el parámetro -w para modificar el valor sobre la marcha:

```
# sysctl -w net.ipv4.conf.all.arp_announce=1
```

7. Esta modificación debe ser definitiva. Modifique el fichero /etc/sysctl.conf y vuelva a cargarlo:

```
# vi /etc/sysctl.conf
```

Añada:

```
net.ipv4.conf.all.arp_announce = 1
```

Y guarde. Vuelve a cargar los nuevos valores:

```
# sysctl -p
```

## 4. Recompilación del núcleo

Objetivo: recompilar el núcleo proporcionado con su distribución.

1. Con el gestor de paquetes de su distribución, instale las fuentes de su núcleo. Esto depende de su distribución. En la mayoría, el paquete se llama kernel-source. Utilice APT, Yum, zypper o una herramienta gráfica que gestione las dependencias.
2. Diríjase a /usr/src/linux y teclee make oldconfig. Eso tiene como efecto que configura el fichero .config de forma idéntica a la configuración actual.

```
# cd /usr/src/linux  
# make oldconfig
```

3. Inicie la configuración texto o gráfico, según su entorno de trabajo, con:

```
# make menuconfig
```

o:

```
# make xconfig
```

Modifique, si lo desea, una opción, por ejemplo para hacer corresponder las optimizaciones del núcleo a su procesador.

4. Inicie la compilación con el comando **make**. Veamos un truco que puede usar si dispone de una máquina potente: make puede simultanear, cuando es posible, la compilación tratando varios ficheros a la vez. Es lo ideal con una máquina multiprocesador (SMP, multicore, HyperThreading, etc.). Utilice la opción -j seguida del número de compilaciones simultáneas:

```
# make -j 2
```

5. Cuando esté terminada la instalación, instale los diferentes componentes, los módulos primero, luego el núcleo:

```
# make modules_install  
# make install
```

Esta última etapa debería crear el initrd y modificar la configuración de GRUB.

6. Verifique en /boot/grub/menu.lst (GRUB) o en /boot/grub/grub.cfg (GRUB2) la presencia del nuevo núcleo, y vuelva a iniciar su máquina para cargar el núcleo.

## 5. Los periféricos y el hardware

Objetivo: manejar los periféricos de cualquier tipo y obtener la información relativa a su hardware.

1. Vaya a /dev y liste todos los periféricos de tipo sd\* o hd\*, según el controlador de sus discos. Liste los números mayor y menor. El número mayor es el mismo para un tipo dado. ¿Qué observa para el número menor? En el caso de un disco SCSI, sólo puede haber 15 particiones, que se numeran para cada disco de sda1 a sda15, sdb1 a sdb15, y así sucesivamente. El número menor del disco es un múltiplo de 16: 0, 16, 32, 48, etc. El número menor de la partición va de 1 a 15, luego de 17 a 31, y así sucesivamente.
2. Aísle la dirección física de su tarjeta gráfica en el bus PCI. Las tarjetas AGP y PCI Express se consideran como si fueran un bus PCI.

```
# lspci | grep -i vga
01:00.0 VGA compatible controller: nVidia Corporation GeForce
8600 GT (rev a1)
```

3. Obtenga más detalles de esta tarjeta. En particular, ¿qué módulo la gestiona? Se puede obtener esta información con -v al especificar únicamente la tarjeta -s de lspci:

```
# lspci -s 01:00.0 -v
01:00.0 VGA compatible controller: nVidia Corporation GeForce
8600 GT (rev a1) (prog-if 00 [VGA controller])
    Subsystem: ASUSTeK Computer Inc. Device 8243
    Flags: bus master, fast devsel, latency 0, IRQ 16
    Memory at fd000000 (32-bit, non-prefetchable) [size=16M]
...
Kernel driver in use: nvidia
Kernel modules: nvidiafb, nvidia
```

4. Liste la información de su procesador. ¿Podría ver si dispone de varios procesadores, o de varios cores, o del HyperThreading?

```
# cat /proc/cpuinfo
```

Para saber si tiene varias CPU, mire los valores de «processor», «cpu cores» y «core id». processor indica el número de procesador. Si hay varios, usted dispone de varios procesadores, SMP clásico, multicore o HyperThreading. Si muestra un valor para «cpu cores» y es superior a 1, su procesador dispone de varios cores, como los Athlons X2 o los Intel Core2 Duo o Quad. En este caso, el «core id» identifica el core actual. Para saber si su máquina gestiona el HyperThreading, mire en los «flags» si «ht» está presente. Si es el caso, en principio el HyperThreading está presente.

5. Conecte un dispositivo USB en su PC. Si usted se encuentra en entorno gráfico, es posible que el gestor de ficheros se abra. ¿Qué mecanismos se ponen en marcha? En un primer tiempo, el núcleo detecta la conexión y carga el módulo USB correspondiente. Se ha creado el periférico básico y se ha generado un evento. El servicio udev detecta el evento y ejecuta las reglas asociadas: modificación, por ejemplo, de los derechos sobre el periférico. Para la interfaz gráfica, hay otro servicio: hal (hardware abstraction layer) es un bus de comunicación entre los diversos elementos. También intercepta el evento y ejecuta otras reglas, esta vez en el espacio del usuario: abre el gestor de ficheros.

# Proceso de inicio

## 1. La BIOS

### a. Papel

La **BIOS** (*Basic Input Output System*) es un software de interfaz entre el hardware y el software a un nivel muy básico. Proporciona el conjunto de las instrucciones básicas utilizadas por el sistema operativo. Provee el nivel de interfaz más bajo a los drivers y periféricos.

La BIOS está presente en una memoria **EEROM** (*Electrical Erasable Programmable Read-Only Memory*) del ordenador. Cuando se enciende el ordenador, o tras un reseteado, se manda una señal llamada *powergood* al microprocesador. Éste activa entonces la ejecución de la BIOS.

La BIOS efectúa un autochequeo del encendido (POST), luego busca los periféricos, en particular los utilizados para iniciar el sistema. A continuación se almacena la información relativa al hardware de manera permanente en una pequeña memoria CMOS alimentada por una batería. Al final de proceso, se selecciona el periférico de inicio.

La BIOS lee y ejecuta el primer sector físico del soporte físico de inicio. Suele tratarse de los 512 primeros bytes del primer disco duro (el MBR) o de la partición activa (la PBR), como se describió en el capítulo Los discos y el sistema de ficheros.

---

 El sucesor de la BIOS se llama UEFI, Unified Extensible Firmware Interface, sucesor a su vez de EFI. Linux soporta UEFI pero el cargador de arranque debe adaptarse en consecuencia. Pocos PC disponen de EFI. Las máquinas Apple no tienen BIOS, sino que disponen de EFI.

---

### b. Ajustes básicos

Cada BIOS es diferente según los fabricantes de tarjetas madre y los editores (AMIBios, Phoenix, Award, etc.). Sin embargo, muchos de los ajustes son idénticos o, en todo caso, se parecen mucho.

La detección de los discos duros y la elección del soporte de inicio se efectúan desde la BIOS. Linux soporta discos IDE, SATA y SCSI. No obstante, es posible que su juego de chips no reconozca el disco SATA. En este caso, la mayoría de las BIOS tienen como opción emular un IDE modificando el controlador de la SATA. Linux los reconocerá como tales. Es recomendable, sin embargo, probar una primera instalación con el soporte nativo de los discos SATA activado.

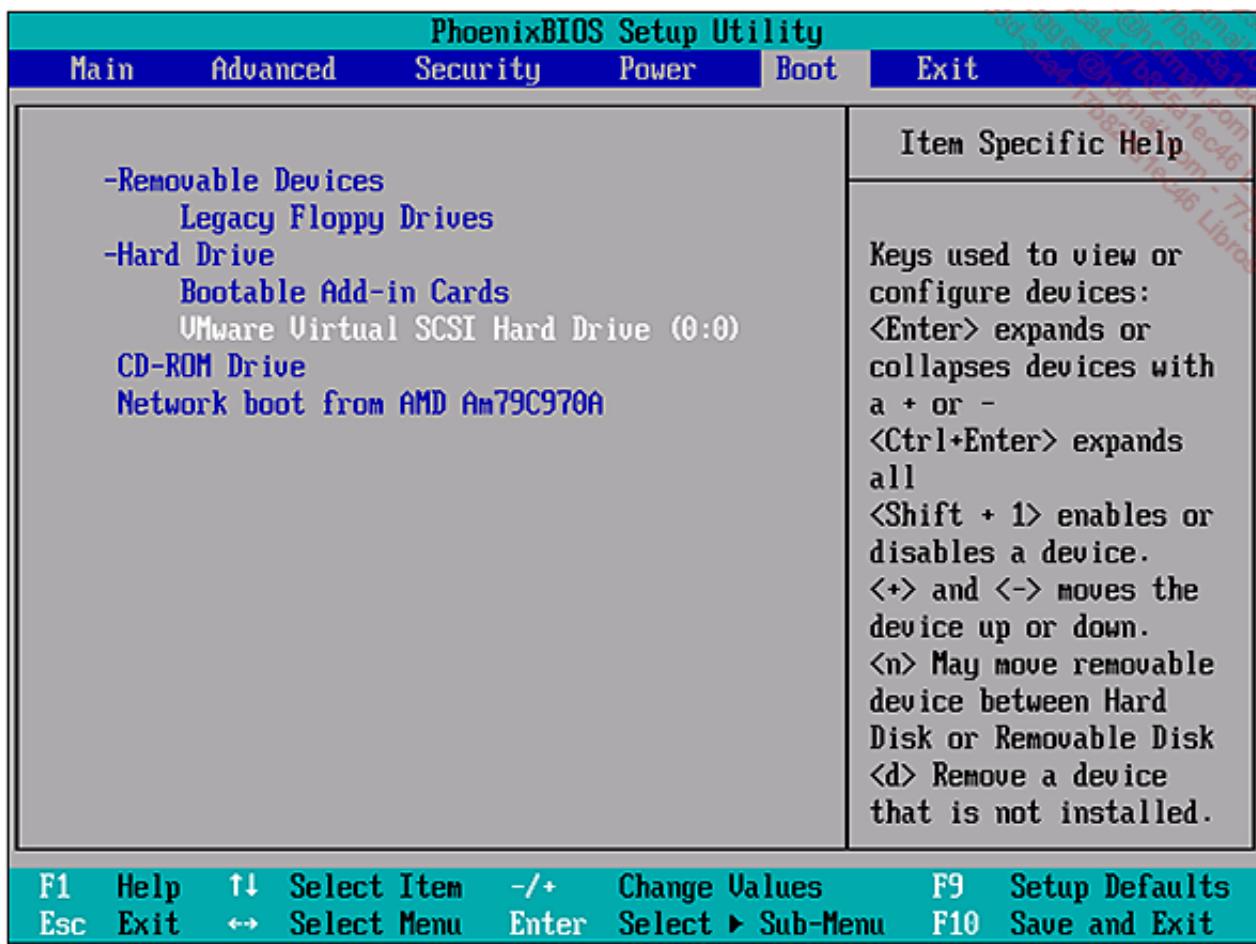
En principio, Linux gestiona correctamente el soporte de los chipsets SATA compatibles **AHCI** (*Advanced Host Controller Interface*), un estándar para las especificaciones públicas. Active esta opción en la BIOS siempre que pueda. En ocasiones aparece con este nombre y, en otras, en modo **nativo**. Si nada funciona, intente el modo **combined**, y luego **legacy IDE**. Encontrará ayuda relativa al SATA en el sitio siguiente: <http://linux-ata.org/faq.html>

Para ejecutar la instalación de Linux desde un CD-Rom o un DVD-Rom, debe modificar el orden de ejecución de manera que arranque primero desde el lector de CD o DVD.

Si su teclado es de tipo USB, o sin cable pero con un adaptador sin cable USB, debe activar el **USB legacy support** (a veces esta función se llama **USB DOS function** o **USB keyboard enable**). Permite activar en el momento del inicio el soporte de los teclados, pero también los soportes de almacenamiento (claves, discos duros, tarjeta de memoria). Esto no impide que el sistema se encargue del USB: una vez iniciado el OS, los drivers USB del núcleo y de los módulos se encargan del USB.

En principio, no tiene por qué tocar la configuración. Evite jugar al aprendiz de hechicero modificando la configuración avanzada del chipset y de otros recursos cuya utilidad no entiende. Sin embargo, con el fin de

de ahorrar recursos, puede desactivar los puertos de la placa base que no utiliza: puerto paralelo, puerto serie, etc.



Pantalla de la BIOS Phoenix que modifica el orden de boot

El overclocking necesita un hardware especial: procesador, placa base, memoria y alimentación deben ser de alta calidad y el PC debe estar bien aireado. El overclocking es fuente de inestabilidad y cuelgues, tanto en Windows como en Linux. En particular, pone a prueba la memoria. Es la principal causa de inestabilidad. Incluso sin overclocking, resulta útil invertir en componentes de calidad.

## 2. El cargador de arranque

La BIOS activa el gestor de arranque inicial (*Initial Program Loader*, IPL) a partir de los primeros 512 bytes del soporte de arranque. En Linux, el cargador está compuesto de dos partes. El cargador inicial de los 512 bytes no contiene bastante código para proponer menús y ejecutar el sistema operativo. Carga un segundo gestor basado en un fichero de configuración.

Este segundo gestor está provisto de una interfaz para ejecutar un sistema operativo de entre una selección dada. Puede aprovechar este gestor para pasárselos parámetros al núcleo Linux y al proceso init.

La BIOS sólo interviene en el inicio de la máquina, durante la utilización del cargador de inicio y en las primeras etapas del carga del núcleo. Luego, no sirve para nada más. El núcleo dispone de sus propias funciones de detección, aunque se apoye en la configuración de la BIOS. En efecto, ésta, bajo la plataforma Intel, se ejecuta en modo real, y Linux, en modo protegido.

### 3. GRUB

#### a. Configuración

En la mayoría de las distribuciones Linux, el cargador por defecto se llama **GRUB** (*Grand Unified Bootloader*). Se puede configurar en gran medida, en particular la protección mediante contraseña encriptada. Cuenta con un intérprete de comandos y con una interfaz gráfica. GRUB guarda las configuraciones en un fichero de texto y no es necesario volver a instalar GRUB tras cada modificación.

A continuación presentamos un ejemplo de configuración a partir del supuesto de que la primera partición del primer disco es /boot y que la segunda contiene una instalación de Windows.

```
timeout=10
default=0
title Red Hat
    root (hd0,0)
    kernel /vmlinuz-2.6.12-15 ro root=LABEL=/
    initrd /initrd-2.6.12-15.img
title Windows XP
    rootnoverify (hd0,1)
    chainloader +1
```

Esta tabla le muestra la sintaxis general de un fichero GRUB:

Parámetro GRUB	Significado
timeout	Número de segundos antes del arranque por defecto.
default n	Arranque por defecto (0=1º título, 1=2º título, etc.).
gfxmenu	Ruta hacia un menú gráfico.
title xxxx	Principio de una sección, entrada del menú de GRUB.
root(hdx,y)	Se especificarán todos los accesos más abajo a partir de esta partición (ver el significado más adelante). Aquí, hd0,0 representa la primera partición del primer disco detectado por la BIOS. Es la partición /boot.
kernel	El nombre de la imagen del núcleo de Linux, seguido de sus parámetros. La / no indica la raíz del sistema de ficheros, sino la de (hd0,0), por lo tanto /boot/vmlinuz...
initrd	Initial ramdisk. Imagen de disco en memoria que contiene la configuración y drivers iniciales y que el sistema sustituirá cuando cargue los discos definitivos.
rootnoverify	La raíz especificada. No debe montarse con GRUB (no soporta NTFS).
chainloader +1	Inicia el primer sector de la raíz especificado más arriba.

Damos el significado de los nombres de periféricos en GRUB.

- (fd0): primer lector de disquetes detectado por la BIOS (/dev/fd0 en Linux).
- (hd0,0): primera partición en el disco duro detectado por la BIOS, ya sea IDE o SCSI (/dev/hda1 o /dev/sda1 según los casos).
- (hd1,4): quinta partición en el segundo disco duro detectado por la BIOS (/dev/hdb5 o /dev/sda5).

## b. Instalación

La configuración de **GRUB** reside en `/etc/grub.conf` o `/boot/grub/menu.lst` (el primero es un vínculo al otro). El binario GRUB puede estar instalado en el **MBR** (*Master Boot Record*, los primeros 512 bytes de un disco) o un **PBR** (*Partition Boot Record*, los primeros 512 bytes de una partición).

Para instalar o desinstalar GRUB en caso de que MBR esté corrompido, por ejemplo en `/dev/sda`, utilice el comando **grub-install**:

```
# /sbin/grub-install /dev/sda
```

## c. Arranque y edición

En el momento del inicio de GRUB, se visualiza un menú. Puede ser gráfico o textual, según la configuración. Tiene que elegir una imagen de inicio entre las propuestas con las flechas de dirección. Pulsando la tecla [Intro], ejecuta la imagen seleccionada.

Puede editar los menús directamente para modificar por ejemplo los parámetros pasados al núcleo de Linux o init. En este caso, seleccione una entrada de menú y pulse la tecla **e** (edit). Aquí se visualizan todas las líneas de la sección. Puede pulsar:

- **e**: para editar la línea (completarla);
- **d**: para suprimir la línea;
- **o**: para añadir una línea;
- **b**: para iniciar la imagen (booter).

Por ejemplo, para iniciar en modo emergencia (emergency):

- Vaya a la línea Linux o Red Hat y pulse **e**.
- Vaya a la línea kernel y pulse **e**.
- Al final de la línea añada 1 o Single y pulse [Intro].
- Teclee **b**.

Puede acceder también a un intérprete de comandos pulsando [Esc]. ¡Cuidado! Sólo se reconocen los comandos GRUB.

# 4. GRUB2

## a. GRUB2, el sustituto de GRUB

GRUB2 es el sucesor de GRUB y se ha reescrito desde cero. De su predecesor sólo conserva el nombre. Ya es el cargador por defecto de algunas distribuciones, entre las cuales hay Ubuntu desde la versión 9.10 o Debian 6. Está dotado de una interfaz gráfica y es modular, compatible con varias arquitecturas (BIOS, EFI, Raid, etc.) y dispone de un modo de recuperación. Sin embargo, su configuración es un poco más difícil de

comprender.

Algo importante que hay que recordar siempre es que la mayor parte de la configuración es automática. Los componentes de GRUB2 detectarán automáticamente la presencia de nuevos núcleos de Linux y de otros sistemas operativos como Windows, gracias al comando **os-prober**. Esta detección no se realiza en el arranque, sino en la ejecución de un comando de actualización.

Su instalación es idéntica a la de la primera versión de GRUB. Se puede migrar de GRUB a GRUB2 utilizando el comando **upgrade-from-grub-legacy**: el archivo menu.lst pasará a estar controlado y se aplicará la nueva configuración.

## b. Configuración

Los archivos de GRUB2 siempre se encuentran en **/boot/grub**. Los nombres de módulos acaban con el sufijo .mod. Hay muchos, especialmente para la gestión del arranque en un gran número de sistemas de ficheros o de tipos de particiones.

La configuración de las entradas de menú está en **/boot/grub/menu.cfg**. Sin embargo, no debe modificar este archivo a mano, ya que debe generarse con el comando **update-grub**.

En la ejecución de **update-grub**, se analizarán varios archivos para generar la configuración:

- **/boot/grub/device.map**, si existe, para la correspondencia entre los nombres de los discos GRUB y Linux.
- **/etc/default/grub**, que contiene los parámetros por defecto de GRUB.
- Todos los archivos albergados en **/etc/grub.d/**, por orden de lista, que son los scripts que permiten generar automáticamente los menús mostrados en el arranque.

## Actualización de GRUB

Cada modificación de estos archivos (o cada vez que se añada un nuevo núcleo o script) debe seguirse inevitablemente de la ejecución del comando **update-grub**:

```
# update-grub
Generating grub.cfg ...
Found linux image: /boot/vmlinuz-2.6.38-8-generic
Found initrd image: /boot/initrd.img-2.6.38-8-generic
Found linux image: /boot/vmlinuz-2.6.35-28-generic
Found initrd image: /boot/initrd.img-2.6.35-28-generic
Found memtest86+ image: /memtest86+.bin
Found Windows 7 (loader) on /dev/sda1
done
```

## device.map

El archivo device.map puede editarse manualmente o generarse automáticamente con el comando **grub-mkdevicemap**. Si el archivo device.map está ausente, la numeración de discos depende del orden de detección de éstos por la BIOS.

```
# grub-mkdevicemap
# cat /boot/grub/device.map
(fd0) /dev/fd0
(hd0) /dev/disk/by-id/ata-STM3500418AS_9VM2LMWK
(hd1) /dev/disk/by-id/ata-WDC_ED5000AAC5-00G8B1_WD_WCAUK0742110
(hd2) /dev/disk/by-id/ata-WDC_WD10EADS-00L5B1_WD-WCAU4A320407
```

## **/etc/default/grub**

El archivo **/etc/default/grub** contiene variables que definen las opciones de GRUB2 y opciones por defecto de los núcleos Linux. A continuación se comentan algunas entradas:

<b>Variable GRUB</b>	<b>Significado</b>
GRUB_DEFAULT	Entrada del menú seleccionada por defecto (primera=0).
GRUB_HIDDEN_TIMEOUT	Considerando una duración en segundos; si esta variable está presente el menú se oculta, permitiendo arrancar en el sistema predeterminado si no es que hay uno solo. Durante este intervalo de tiempo si el usuario pulsa [Esc] o [Shift] aparecerá el menú por pantalla.
GRUB_TIMEOUT	Duración de la visualización del menú en segundos antes de que arranque la entrada por defecto.
GRUB_HIDDEN_TIMEOUT_QUIET	Define si se visualiza o no el contador de timeout si GRUB_HIDDEN_TIMEOUT está definido: true o false.
GRUB_DISTRIBUTOR	Línea de comandos que define la generación automática del texto en la entrada del menú.
GRUB_CMDLINE_LINUX_DEFAULT	Parámetros pasados al núcleo Linux, únicamente para las entradas "normales", no las de recuperación.
GRUB_CMDLINE_LINUX	Parámetros por defecto para todos los núcleos LINUX.
GRUB_GFXMODE	Resolución utilizada por GRUB2 en modo gráfico, que puede modificarse por un valor devuelto por el comando <b>vbeinfo</b> .
GRUB_TERMINAL	Sin comentar y con el valor <b>console</b> , se activa el modo texto.

A continuación un ejemplo:

```
GRUB_DEFAULT=0
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash nomodeset"
GRUB_CMDLINE_LINUX=""
GRUB_GFXMODE=640x480
```

## **Construcción de los menús**

Los menús se construyen con los scripts albergados en **/etc/grub.d/**:

```
# ls -l /etc/grub.d/*
-rwxr-xr-x 1 root root 6658 2011-05-02 21:04 /etc/grub.d/00_header
```

```
-rwxr-xr-x 1 root root 5522 2011-04-21 15:19 /etc/grub.d/05_debian_theme  
-rwxr-xr-x 1 root root 6291 2011-05-02 21:43 /etc/grub.d/10_linux  
-rwxr-xr-x 1 root root 5233 2011-04-21 15:31 /etc/grub.d/20_linux_xen  
-rwxr-xr-x 1 root root 1588 2010-09-24 20:16 /etc/grub.d/20_memtest86+  
-rwxr-xr-x 1 root root 7119 2011-04-21 15:31 /etc/grub.d/30_os-prober  
-rwxr-xr-x 1 root root 214 2010-10-06 14:22 /etc/grub.d/40_custom  
-rwxr-xr-x 1 root root 95 2010-10-06 14:22 /etc/grub.d/41_custom  
-rw-r--r-- 1 root root 483 2010-10-06 14:22 /etc/grub.d/README
```

Estos archivos se ejecutarán en el orden de listado (orden alfanumérico de forma predeterminada) de ls. Es por este motivo que empiezan con un valor numérico. Los dos scripts interesantes son **10\_linux** y **30\_os-prober**.

Si examina el primero, encontrará como elemento particular los bucles que buscan los núcleos Linux en /boot y en la raíz, así como los ramdisk iniciales (ver más adelante), para detectar los distintos núcleos Linux y crear automáticamente las entradas asociadas añadiendo los parámetros por defecto.

En el segundo, verá que se ejecuta el comando **os-prober**, que busca el resto de sistemas operativos en el resto de discos y particiones, como Windows, Hurd, BDS, Mac OS X o incluso Linux. Lo puede comprobar usted mismo:

```
# os-prober  
/dev/sdal:Windows 7 (loader):Windows:chain
```

Puede añadir sus propios scripts partiendo de alguna plantilla de las que vienen con el paquete (**40\_custom**, por ejemplo). Lo que devuelve su script se añade automáticamente a grub.cfg, del cual hay que respetar su sintaxis.

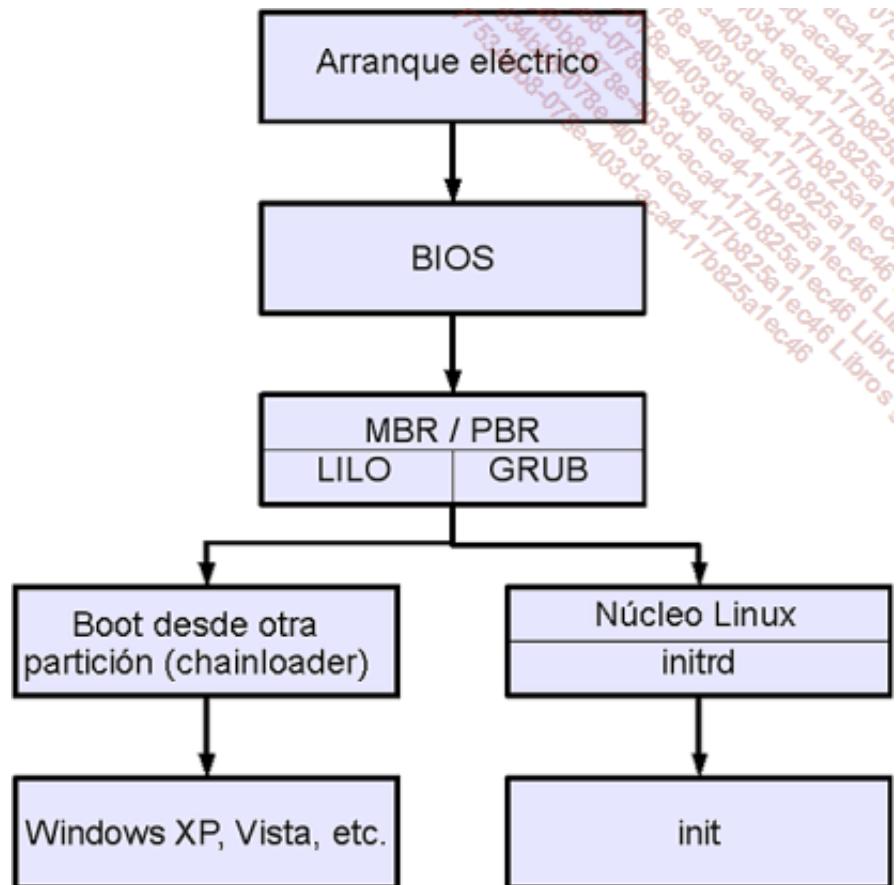
### c. Arranque y edición

El uso de GRUB2 en el arranque es muy parecido al de GRUB, pero con algunas sutilezas. Para editar alguna entrada use siempre la tecla **e** y modifique las distintas líneas como anteriormente. El editor se comporta como un miniEmacs. Se tendrá que usar [Ctrl] **x** para arrancar con sus modificaciones.

## 5. Inicialización del núcleo

Durante la carga del núcleo, el monitor le mostrará una tonelada de información. El sistema no la registra en esta etapa, pero en la siguiente, la etapa del init, Linux comienza a escribir registros en el fichero **/var/log/dmesg**.

- Se detecta e inicializa el hardware.
- initrd está cargado, los módulos presentes se han cargado si es preciso.
- El núcleo monta el sistema de ficheros raíz en modo de sólo lectura.
- Crea la primera consola.
- Se ha iniciado el primer proceso (en general init).



*Esquema de la secuencia de inicio*

Existen otros cargadores, entre los cuales el más usado hasta ahora era **LILO** (*Linux Loader*). Sin embargo, estos últimos años, GRUB lo ha sustituido casi por completo como consecuencia de sus numerosas limitaciones.

# init System V

## 1. Funciones

El programa init es el primer proceso iniciado y el último que se para dentro del sistema, y tiene como misión ejecutar todas las demás tareas. El papel inicial de init consiste en iniciar y parar todos los servicios. init ejecutará las diferentes tareas iniciales necesarias para el buen funcionamiento de Linux mediante la ejecución de varios comandos y scripts.

Una vez iniciado el sistema y ejecutados los servicios, init sigue activo para gestionar los cambios de estado de los procesos que controla y de los niveles de ejecución.

El programa init puede cambiar de una distribución a otra. En la mayoría de las distribuciones profesionales o mayores (Mandriva, Red Hat/Fedora, openSUSE, Debian, etc.), el principio sigue siendo más o menos el mismo: init de tipo System V (basado en la noción de niveles de ejecución). La distribución Ubuntu utiliza Upstart, que gestiona las prioridades, los eventos y dependencias entre los servicios, pero respeta el mismo principio. La distribución Slackware utiliza otro mecanismo inspirado en BSD. Finalmente, la distribución Fedora utiliza desde la versión 15 systemd otro reemplazo de init, de diseño moderno y disponible únicamente para Linux. Parece que con el tiempo systemd también reemplazará init.

Sin embargo, la versión System V de init aún goza de días de esplendor: no sólo la mayoría de las distribuciones Linux en el ámbito empresarial la siguen usando, sino que además su principio de diseño y su configuración son casi idénticos a las versiones Unix propietarias.

El proceso **init** es el padre de todos los procesos. Siempre tiene el PID 1. Su configuración está en el fichero **/etc/inittab**. Si este fichero está corrompido o es inutilizable, habrá que arrancar el sistema en modo single (S, s, 1 Single) y arreglarlo, o a lo peor arrancar desde un soporte externo o un disco de emergencia. Es un fichero central del sistema operativo.

## 2. Nivel de ejecución

Un nivel de ejecución, o runlevel, corresponde al estado en el cual se encuentra Unix/Linux. Init controla este estado. Cada estado dispone de su propia configuración (o por inittab, o por scripts llamados initscripts). Por ejemplo, se puede utilizar un nivel de ejecución para arrancar Unix en modo monousuario, en multiusuarios, con o sin red, con o sin modo gráfico. El administrador puede personalizar todos los niveles. Por convención, en las distribuciones Red Hat/Fedora, Mandriva, openSUSE y asociadas, se suelen definir estos niveles como a continuación:

Nivel	Efecto
0	Halt: para el sistema operativo, apaga la máquina.
1	Modo monousuario utilizado para el mantenimiento, modo consola.
2	Multiusuario, sin red, consola.
3	Multiusuario, con red, consola.
4	ídem que el 3, a conveniencia del administrador.
5	Multiusuario, con red, con entorno gráfico X Window.

6	Reboot: reinicio de la máquina.
S,s	Single user mode, el modo más bajo en caso de problema.

Los niveles 7 a 9 son perfectamente válidos, pero no se utilizan por defecto. El nivel de ejecución se sitúa por defecto en `/etc/inittab` en la línea initdefault.

`id:5:initdefault:`

Sustituya 5 por el nivel deseado en el momento del arranque.

 Un reto en ciertos ejercicios eliminatorios de certificaciones consiste en crear una situación de avería en la cual el valor por defecto está en 0 o 6. En el primer caso, la máquina se apaga en cuando se ejecuta init. En el otro, arranca en bucle...

La distribución Debian (y las distribuciones que derivan de ella) considera también los niveles 2 a 5 como multiusuario, pero no establece diferencias entre estos niveles. Por defecto, se arranca en el nivel 2, donde se inicia todo, incluso en su caso la interfaz gráfica.

Como se puede modificar y volver a configurar completamente cada nivel, es posible volver a definirlo todo, y por lo tanto, hacer que una Debian se inicie como una Red Hat, y viceversa. Por motivos de conformidad y soporte, considere importante seguir el «estándar» de la distribución que usted utiliza.

### 3. /etc/inittab

Se define el comportamiento del proceso init y de los runlevels en el fichero `/etc/inittab`. La sintaxis de una línea es la siguiente:

`Id:[niveles]:acción:comando`

Campo	Descripción
Id	Identificador de línea sobre cuatro caracteres (en Linux con getty/mingetty: número de terminal).
Niveles	Indica si se debe tener en cuenta el comando para el nivel requerido. Corresponde a la lista de los niveles sin separador.
Acción	Tipo de acción a efectuar según las circunstancias para esta línea.
Comando	El comando a ejecutar con sus parámetros y las redirecciones.

La acción es muy importante, ya que define las actividades de init durante el arranque y cambio de nivel.

Las principales se presentan a continuación:

Acción	Significado
initdefault	Define el nivel por defecto durante el boot y el inicio de init.
sysinit	Se ejecuta una única vez durante el arranque del sistema.

boot	Ídem, pero después de sysinit.
bootwait	Ídem, pero init espera el final de la ejecución del comando antes de continuar leyendo el fichero inittab.
off	Se ignora la línea.
once	Se ejecuta el comando a cada cambio de nivel para los niveles especificados.
wait	Ídem, pero init espera el final de la ejecución antes de proseguir.
respawn	El comando se ejecuta para los niveles correspondientes. Si el proceso termina, vuelve a arrancarse de nuevo automáticamente. Es el caso para los terminales si un usuario se desconecta de ellos.
powerwait	Inicia el comando si el servidor es alimentado mediante una fuente de emergencia (UPS).
powerfail	Ídem, pero sin esperar el final de ejecución del comando.
powerokwait	Se ejecuta el comando cuando se restablece la corriente.
powerfailnow	Comando de último recurso cuando la alimentación de emergencia está casi vacía.
ctrlaltdel	Init recibe una señal SIGINT procedente de una secuencia [Alt][Ctrl][Supr].

Damos un ejemplo procedente de una instalación openSUSE 10.3:

```
# Nivel de ejecución a 5 (multiusuario gráfico)
id:5:initdefault:

# Primer script ejecuta el arranque
si::bootwait:/etc/init.d/boot

# Gestión de los servicios por nivel de ejecución
l0:0:wait:/etc/init.d/rc 0
l1:1:wait:/etc/init.d/rc 1
l2:2:wait:/etc/init.d/rc 2
l3:3:wait:/etc/init.d/rc 3
l4:4:wait:/etc/init.d/rc 4
l5:5:wait:/etc/init.d/rc 5
l6:6:wait:/etc/init.d/rc 6

# Caso del modo single, consola de emergencia para root
ls:S:wait:/etc/init.d/rc S
~~:S:respawn:/sbin/sulogin

# Acción en Alt+Ctrl+Del
ca::ctrlaltdel:/sbin/shutdown -r -t 4 now

# Qué hacer en caso de corte de la corriente
pf::powerwait:/etc/init.d/powerfail start
pn::powerfailnow:/etc/init.d/powerfail now
po::powerokwait:/etc/init.d/powerfail stop
```

```
# Arranque de las consolas virtuales Alt+Fx
1:2345:respawn:/sbin/mingetty --noclear tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

## 4. Cambio de nivel

También puede cambiar de nivel después de arrancar la máquina con el comando **/sbin/init o /sbin/telinit**, al ser este último un simple vínculo simbólico a init. El comando siguiente pasa al nivel 5.

```
# telinit 5
```

Se pueden especificar los valores q, u o -t:

- Q o q: init vuelve a leer el fichero **/etc/inittab**, si se ha modificado, y corrige sus tablas internas.
- U o u: init vuelve a iniciarse sin leer inittab y sin cambiar de nivel. Si se han añadido o suprimido servicios del nivel en curso, init tiene en cuenta la modificación.
- -t: cuando init termina la parada de los servicios (o más bien cuando lo hace el script rc; ver más adelante), init manda la señal SIGTERM a todos los procesos restantes, les pide que se terminen correctamente, espera el número de segundos especificado (5 por defecto), luego manda SIGKILL.

El nivel de ejecución es visible con el comando **/sbin/runlevel**. El primer valor devuelto corresponde al nivel que precede el nivel actual. Una N significa que no hay un nivel precedente. El segundo valor es el nivel actual.

```
# runlevel
N 5
```

## 5. Configuración del sistema básico

Sea cual sea el nivel de ejecución especificado por defecto, init inicia siempre el comando asociado a las acciones sysinit, bootwait o boot de **/etc/inittab** en el momento de arrancar el sistema. La acción sysinit es la primera.

- En Red Hat: si::sysinit:/etc/rc.d/rc.sysinit
- En openSUSE: si::bootwait:/etc/init.d/boot
- En Debian: si::sysinit:/etc/init.d/rcS

En Red Hat, es un script único monolítico que se encarga de toda la configuración básica. En Debian, el script llama todos los scripts del nivel S (single). En openSUSE, el script instala lo estrictamente necesario y luego ejecuta el contenido de **/etc/rc.d/boot.d**, que establece el resto de la configuración básica.

En todos los casos, se ejecutan las tareas siguientes más o menos en este orden:

- Configuración de los parámetros del núcleo presentes en **/etc/sysctl.conf** (p. ej.: IP Forwarding).
- Instalación de los ficheros periféricos (/dev vía udev, por ejemplo).
- Configuración del reloj del sistema.
- Carga de las tablas de caracteres del teclado.
- Activación de las particiones de intercambio SWAP.

- Definición del nombre de anfitrión.
- Control y montaje del sistema de ficheros raíz (en lectura-escritura esta vez).
- Añadido de los periféricos RAID, LVM o ambos. Esto se puede ya instalar durante la carga de inittab.
- Activación de las cuotas de disco.
- Control y montaje de los otros sistemas de ficheros.
- Limpieza de los bloqueos (stale locks) y de los ficheros PID en caso de parada brusca.

En algunas distribuciones es posible interactuar con init. Al principio del boot, después del arranque de init, se le puede requerir que pulse la letra i, luego que conteste sí o no a las diferentes acciones.

## 6. Nivel de ejecución System V

### a. rc

El script `/etc/init.d/rc` coge como parámetro el nivel de ejecución por defecto según la línea **initdefault** de `/etc/inittab` o el parámetro especificado durante la llamada manual de los comandos **init** o **telinit**. El script rc inicializa el nivel de ejecución deseado y es responsable del inicio y de la parada de los servicios asociados cuando el nivel de ejecución cambia.

```
l1:1:wait:/etc/init.d/rc 1
l2:2:wait:/etc/init.d/rc 2
l3:3:wait:/etc/init.d/rc 3
l4:4:wait:/etc/init.d/rc 4
l5:5:wait:/etc/init.d/rc 5
l6:6:wait:/etc/init.d/rc 6
```

Se analizan los servicios en cada nivel de ejecución. Durante el paso de un nivel a otro, y sea cual sea el orden (del 2 al 5, del 5 al 3, etc.) el script rc compara entre el antiguo y el nuevo nivel los servicios que se deben detener o iniciar. Si un servicio es común a los dos niveles, lo mantiene. Si se debe iniciar un nuevo servicio en el nuevo nivel, lo hace. Si se debe parar un servicio porque está ausente del nuevo nivel, lo para.

---

 Este funcionamiento, estándar a todas las distribuciones Linux de tipo System V, no es común a todos los Unix. HP-UX (un Unix de HP) considera que debe haber una progresión constante en los niveles, pasando sucesivamente del 1 al 3 (1 luego 2 luego 3) y cargando de manera sucesiva los servicios. Parado, baja hasta el nivel 0 y termina los servicios sucesivamente. La diferencia es importante: no compara los niveles y no efectúa parada/reinicio entre cada nivel...

---

## 7. Gestión de los niveles y de los servicios

### a. Servicios en init.d

El nivel de ejecución define los servicios que se deben iniciar para este nivel. Le corresponde al script rc cargar los servicios. Se controlan los servicios (inicio, parada, reinicio, estatus, etc.) mediante scripts presentes en el directorio `/etc/init.d`.

```
# cd /etc/init.d
# ls -l
-rwxr-xr-x 1 root root 1128 ago  9  2004 acpid
```

```

-rwxr-xr-x 1 root root 834 sep 28 2004 anacron
-rwxr-xr-x 1 root root 1429 jun 22 2004 apmd
-rwxr-xr-x 1 root root 1176 abr 14 2006 atd
-rwxr-xr-x 1 root root 2781 mar 5 2007 auditd
-rwxr-xr-x 1 root root 17058 sep 5 2007 autofs
-rwxr-xr-x 1 root root 1368 feb 2 2007 bluetooth
-rwxr-xr-x 1 root root 1355 may 2 2006 cpuspeed
-rwxr-xr-x 1 root root 1904 jul 16 2007 crond
-rwxr-xr-x 1 root root 2312 oct 30 13:46 cups
...

```

Para cada nivel de ejecución *n*, existe un directorio *rcn.d* que contiene vínculos simbólicos (atajos) hacia los servicios presentes en */etc/init.d* que se quieren iniciar o parar. Este directorio se puede encontrar en diferentes lugares según la distribución:

- Red Hat: */etc/rc.d/rcn.d* con vínculos en */etc/rcn.d*
- openSUSE: */etc/init.d/rcn.d*, sabiendo que */etc/rc.d* apunta a */etc/init.d*
- Debian: */etc/rcn.d*

El prefijo del nombre de cada vínculo define su orden de ejecución o de parada. El nombre está con la forma siguiente:

[SK]nnservicio

- **S**: start.
- **K**: kill (stop).
- **nn**: orden numérico de ejecución o parada (00=primero, 99=último).
- **servicio**: nombre del servicio.

Por ejemplo, el vínculo S10network indica que se iniciará el servicio network, responsable de la instalación de la red, en orden 10, después de los S01, S05, etc., pero antes de los S11, S15, S20, etc.

```

# ls -1 S*
S00microcode_ctl
S01sysstat
S02lvm2-monitor
S05kudzu
S06cpuspeed
S08iptables
S09isdn
S09pcmcia
S10network
S12syslog
S13irqbalance
S13portmap
S14nfslck
S15mdmonitor
S18rpcidmapd
...

```

Cuando se ejecuta rc, primero lista todos los vínculos que comienzan por K\* usando un bucle for. Luego hace lo mismo para S\*, y esta vez inicia los servicios. Presentamos una parte del fichero rc para entender mejor la secuencia de inicio:

```

# prueba de la existencia de /etc/rcn.d
if [ -d /etc/rc${level}.d ]

```

```

then
# Lista todos los scripts que empiezan por S en este directorio
for i in /etc/rc${level}.d/S*
do
    # el script existe y no está vacío: se ejecuta
    if [ -s ${i} ]
    then
        sh ${i} start
    fi
done
fi

```

## b. Control manual de los servicios

### Mediante scripts

Se pueden iniciar los servicios en todos los casos individualmente o con la ayuda de herramientas según la distribución. El primer método es el único por defecto en Debian.

Cada servicio presente en /etc/init.d acepta al menos dos parámetros:

- **start**: el servicio se inicia.
- **stop**: el servicio se para.

Si usted desea iniciar o parar el servicio sshd (servidor ssh) manualmente:

```

# /etc/init.d/sshd start
Starting SSH daemon                                         done
# /etc/init.d/sshd stop
Shutting down SSH daemon                                     done

```

Algunos servicios pueden aceptar otros parámetros:

```

# /etc/init.d/sshd
Usage: /etc/init.d/sshd {start|stop|status|try-
restart|restart|force-reload|reload|probe}

```

**status**: facilita el estado del servicio (iniciado o no). Según los servicios, se puede facilitar una información adicional.

- **probe**: indica si es necesario cargar la configuración; si, por ejemplo, se han modificado ficheros de configuración.
- **reload / force-reload**: indica al servicio que vuelva a leer su configuración (vía una señal 1).
- **restart**: espera y vuelve a iniciar el servicio, sea cual sea el final de la parada.
- **try-restart**: para y vuelve a iniciar el servicio solamente en caso de parada.

---

 La distribución openSUSE crea vínculos simbólicos **rc<servicio>** que permiten insertar **rcsshd**, por ejemplo para el control manual de los servicios.

---

### Mediante el comando service

El comando **service** está disponible en Red Hat y openSUSE. Permite simplemente prescindir de la ruta hacia el script de inicio del servicio y utilizar simplemente su nombre:

```
# service sshd stop
Shutting down SSH daemon                                done
# service sshd start
Starting SSH daemon                                     done
```

Para controlar la configuración de los servicios de System V lanzados por init, no se aconseja hacerlo todo a mano, sino más bien utilizar las herramientas del sistema correspondiente cuando existan, tanto en modo texto o como gráfico.

### c. Modificación de los niveles de ejecución

#### **Red Hat y openSUSE**

En Red Hat/Fedora y openSUSE, el comando **chkconfig** permite añadir, suprimir, activar o desactivar scripts, por nivel de ejecución. Este comando es muy práctico para configurar los servicios porque sabe gestionar tanto los servicios System V como los servicios xinetd.

```
chkconfig [opción] [servicio]
```

 Aunque la sintaxis sea idéntica en las dos distribuciones, chkconfig no funciona de la misma manera. En Red Hat, se inserta una línea especial al principio del script que indica a chkconfig los parámetros por defecto (runlevels, posiciones de inicio y parada). En openSUSE, chkconfig es un frontend para el comando **insserv**. Este último aprovecha también el encabezamiento de los scripts, pero de manera más compleja (gestiona la ordenación y el paralelismo, por ejemplo).

Presentamos las primeras líneas del script de inicio de servicio sshd en Red Hat. El script se inicia y se para en los niveles 2, 3, 4 y 5. Se ejecuta en posición 55 (S55sshd) y se para en posición 25 (K25sshd).

```
# chkconfig: 2345 55 25
# description: OpenSSH server daemon
```

Veamos lo mismo en openSUSE. chkconfig e insserv gestionan ellos mismos el orden de inicio y parada gracias a los campos Required-Start y Required-Stop. Se deben iniciar los servicios redes y remote\_fs antes de sshd. El servicio se inicia a niveles 3 y 5 y se para a los niveles 0 (parada), 1 (single user), 2 (sin red) y 6 (reboot).

```
### BEGIN INIT INFO
# Provides: sshd
# Required-Start: $network $remote_fs
# Required-Stop: $network $remote_fs
# Default-Start: 3 5
# Default-Stop: 0 1 2 6
# Description: Start the sshd daemon
### END INIT INFO
```

Damos la lista de las opciones de chkconfig:

- **--list**: lista del conjunto de la configuración.
- **--list servicio**: la configuración de un servicio dado.
- **--add servicio**: añade el servicio indicado en la configuración System V.
- **--del servicio**: suprime el servicio de la configuración System V.
- **--level xxx servicio on/off**: activa o desactiva el servicio para los niveles de ejecución indicados.

```

# chkconfig -list
rwhod    0:parada 1:parada  2:parada  3:parada  4:parada  5:parada  6:parada
atd      0:parada 1:parada  2:parada  3:funciona 4:funciona 5:funciona 6:parada
snmpd   0:parada 1:parada  2:funciona 3:funciona 4:funciona 5:funciona 6:parada
ntpdd   0:parada 1:parada  2:funciona 3:funciona 4:funciona 5:funciona 6:parada
keytable 0:parada 1:funciona 2:funciona 3:funciona 4:funciona 5:funciona 6:parada
syslog  0:parada 1:parada  2:funciona 3:funciona 4:funciona 5:funciona 6:parada
...
...
# chkconfig --list smb
smb      0:parada 1:parada  2:parada  3:parada  4:parada  5:parada  6:parada
# chkconfig --level 35 smb on
# chkconfig --list smb
smb      0:parada 1:parada  2:parada  3:funciona 4:parada  5:funciona 6:parada
# /sbin/chkconfig --add httpd

```

 **chkconfig** no inicia ningún servicio. Sólo configura los niveles de ejecución. Para iniciar un servicio, se utilizará el script asociado o el comando **service**.

---

### **En Debian**

El comando **update-rc.d** crea los vínculos necesarios en los distintos directorios **rcn.d**, un poco como **chkconfig**, pero de manera más «bruta»: los scripts no tienen la información particular en su encabezamiento y el comando sólo funciona a nivel del sistema de ficheros, por lo que instala los diferentes vínculos según sus indicaciones.

Veamos dos ejemplos de sintaxis. El primer inscribe un servicio con parámetros por defecto. En este caso, se configura el servicio para ejecutarse en los niveles de 2 a 5 y pararse a los niveles 0, 1 y 6. La posición de parada/relanzamiento está a 20.

```
# update-rc.d ssh defaults
```

En el segundo ejemplo, se inserta el servicio con opciones completas. Al iniciarse, el servicio está en posición 10 en los niveles 3, 4 y 5. Parado, el servicio está en la posición 5 en los niveles 0, 1 y 6. No olvide los puntos.

```
# update-rc.d ssh start 10 3 4 5 . stop 05 0 1 6 .
```

El parámetro **remove** suprime los vínculos de los diversos directorios. Sin embargo, el propio script/**/etc/init.d/xxx** asociado ya no debe de existir. En caso contrario, utilice el parámetro **-f** para forzar la supresión de los vínculos (el propio script sigue en su sitio).

```
# update-rc.d -f ssh remove
```

## 8. Consolas virtuales

Las consolas virtuales permiten obtener terminales virtuales en una máquina. Se definen en **/etc/inittab**. Están disponibles mediante los periféricos **/dev/ttyn**, donde n es el número de consola.

No se suele instalar ni iniciar la capa gráfica en los servidores de empresa. Usted accede al servidor o bien directamente (o por kvm), o bien por la red (ssh por ejemplo) o mediante soluciones integradas (puertos de administración del servidor, como ILO en las máquinas HP).

- Pase de una consola a la otra con la secuencia de teclas [Alt][Fn] (p. ej.: [Alt][F2]) desde la consola o [Ctrl][Alt][Fn] desde X Window.

- Utilice las teclas [Alt][Flecha derecha] y [Alt][Flecha izquierda] para pasar a la consola siguiente o a la anterior.
- /dev/ttyn representa la consola virtual n.
- /dev/tty0 representa la consola actual.
- Como hay 12 teclas de función, puede haber por defecto 12 terminales virtuales.
- Sin embargo, se activan «solamente» 6 por defecto.
- Se inicia X Window por defecto en la primera consola disponible, en general la 7.

Se inician las consolas con inittab y los procesos **getty** o **mingetty**. Son las únicas entradas de inittab donde la etiqueta tiene su importancia: corresponde al número de la consola. Observe la utilización de respawn. Como la consola establece la petición de login, luego el shell se sustituye y, cuando se termina, se inicia un proceso mingetty automáticamente para aceptar una nueva conexión.

```
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

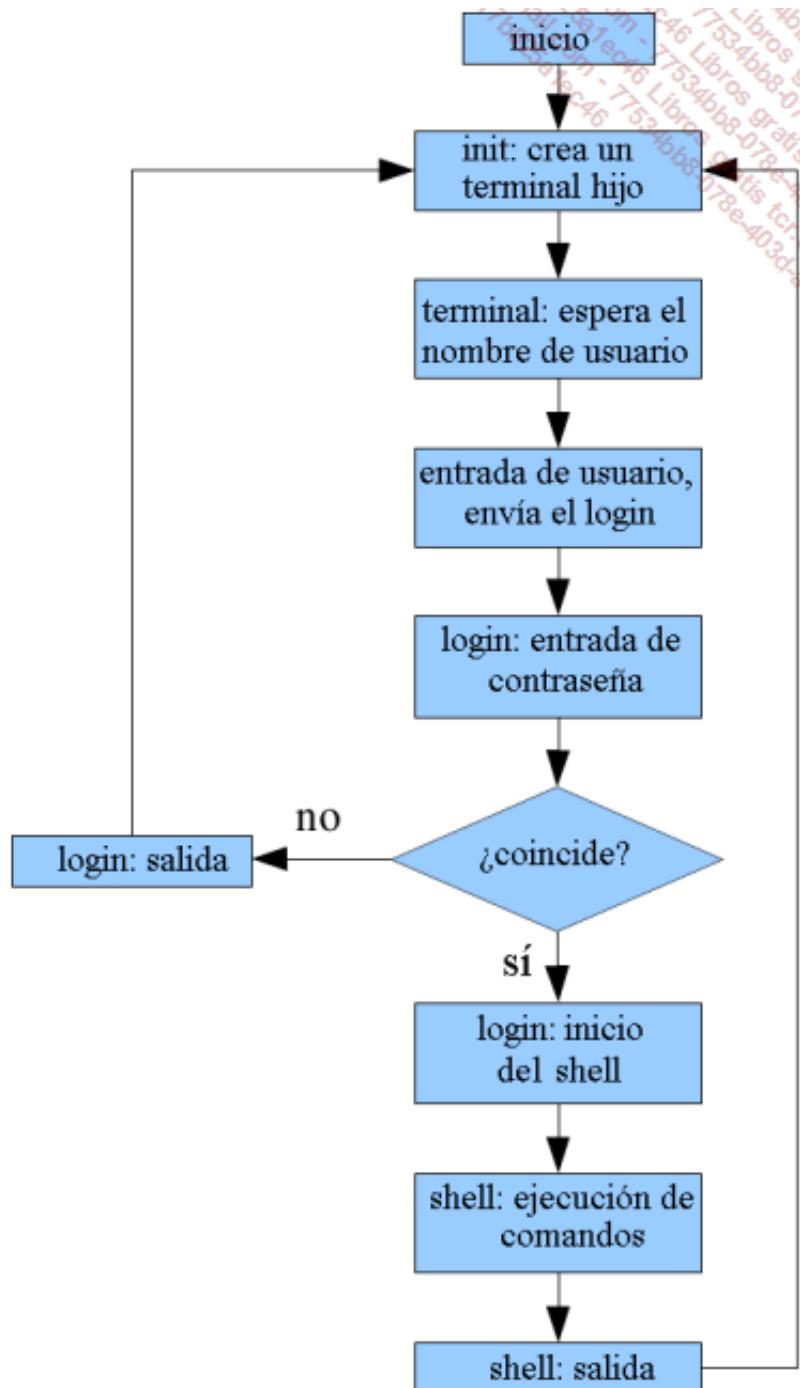
## 9. Los logins

Una vez que init ha iniciado los terminales (getty), un prompt espera la inserción del nombre (el login) del usuario. Antes de este prompt, se visualiza el contenido del fichero **/etc/issue**. Con el nombre validado, el terminal ejecuta el comando **login**, que va a requerir la inserción de la contraseña. Si la contraseña es correcta (verificación en **/etc/passwd** y **/etc/shadow** o utilización de los módulos PAM), entonces login muestra el contenido de **/etc/motd** y ejecuta un shell (siempre definido en **/etc/passwd**).

Observe que getty y login no efectúan un fork: los procesos iniciados no son hijos, pero solapan el proceso actual (API exec). No hay relaciones hijos-padres entre los procesos (getty->login->shell) pero cada uno sustituye al precedente, y el proceso guarda el mismo PID. De esta manera init sabe cuándo se termina una conexión.

Una vez terminada la sesión (fin del shell), init vuelve a iniciar un terminal para una nueva conexión (comando respawn).

getty va a permitir un buen funcionamiento del terminal del usuario: para ello se va a adaptar a los diferentes parámetros de éste (VT100, VT220, XTERM, CONSOLE...). Además, getty puede perfectamente escuchar un puerto serie y soportar una conexión con módem (e iniciar luego una sesión ppp, por ejemplo).



*Secuencia de apertura de sesión vía login*

## 10. Parada

Varios métodos permiten parar correctamente una máquina en Linux. Antes de nada y a título orientativo, init gestiona las paradas con los niveles 0 y 6. En la práctica, los dos son casi idénticos salvo para la última acción.

- Runlevel 0: el ordenador se apaga automáticamente.
- Runlevel 6: el ordenador se apaga y se reinicia de nuevo.

De esta manera, el comando siguiente apaga el ordenador:

```
# init 0
```

Este comando vuelve a iniciar el sistema:

```
# init 6
```

Sin embargo, el comando más correcto, más propio y más seguro para parar el sistema es **shutdown**. Shutdown llama a init, pero acepta parámetros adicionales. Su sintaxis básica es: shutdown <param> <plazo> <mensaje>

Las parámetros son:

Parámetro	Acción
-k	No apaga el sistema, sino que manda el mensaje de apagado a todo el mundo.
-r	Reinicio.
-h	(halt) Parada.
-f	Impide la ejecución de fsck en el boot.
-F	Fuerza la ejecución de fsck en el boot.
-c	Cancela el shutdown sin demora, pero puede enviar un mensaje.

Se puede especificar el plazo de diferentes maneras:

- **hh:mm**: una hora precisa.
- **+m**: en m minutos.
- **now**: un alias para +0, es decir, ahora mismo.

El ejemplo siguiente programa un reinicio para dentro de 10 minutos con un mensaje de aviso.

```
# shutdown -r +10 "Reinicio para mantenimiento en 10 minutos"  
Broadcast message from root (pts/2) (Fri Apr 4 15:00:34 2008):  
Reinicio para mantenimiento en 10 minutos  
The system is going DOWN for reboot in 10 minutes!
```

Ejemplo siguiente cancela el reinicio.

```
# shutdown -c "Mantenimiento cancelado"  
Shutdown cancelled.  
Broadcast message from root (pts/2) (Fri Apr 4 15:02:21 2008):  
Mantenimiento cancelado
```

Se llama a los comandos **reinicio** y **halt** al final de init 6 y 0, respectivamente. Si se llaman a un nivel diferente de 6 o 0, ello equivale a una llamada de shutdown:

- **halt**: shutdown -h
- **reboot**: shutdown -r

# Alternativas a init

## 1. upstart

### a. Fundamentos

upstart es un sustituto del servicio init, inicialmente desarrollado para Ubuntu por un empleado de Canonical, que funciona de forma asíncrona y basada en eventos. Controla el conjunto de servicios, gestiona su inicio y su parada, pero también supervisa su funcionamiento. Por razones más que evidentes, es compatible con init System V: cualquier servicio System V funcionará con upstart. Sin embargo, los servicios upstart no funcionarán con System V. upstart ahora es el init predeterminado de muchas distribuciones y esta migración se ha realizado de forma flexible dada dicha compatibilidad.

Con upstart, los servicios pueden iniciarse de forma asíncrona y, por lo tanto, pueden parallelizarse, y no iniciarse secuencialmente como con init. De este modo, se pueden definir dependencias entre los servicios y optimizar sus inicios y sus paradas. Basado en eventos, upstart podrá, por ejemplo, iniciar los servicios Bluetooth únicamente si una llave USB de este tipo se ha conectado y no de forma inútil y sistemática en el arranque de la máquina.

### b. Archivos

Los archivos de configuración de upstart se encuentran en **/etc/init**. Son archivos de configuración con la sintaxis específica, aunque también pueden contener shell script clásico. A continuación se muestra a modo de ejemplo el archivo de configuración que se encarga del servicio sshd:

```
# ssh - OpenBSD Secure Shell Server
#
# The OpenSSH server provides secure shell access to the system.

Description      "OpenSSH server"

start on filesystem or runlevel [2345]
stop on runlevel [!2345]

respawn
respawn limit 10 5
umask 022

pre-start script
test -x /usr/sbin/sshd || { stop; exit 0; }
test -e /etc/ssh/sshd_not_to_be_run && { stop; exit 0; }
test -c /dev/null || { stop; exit 0; }

mkdir -p -m0755 /var/run/sshd
end script

# if you used to set SSHD_OPTS in /etc/default/ssh, you can change the
# 'exec' line here instead
exec /usr/sbin/sshd -D
```

La sintaxis es comprensible para el administrador. Las directivas **start** y **stop** indican con qué eventos se inicia y se detiene el servicio. En este caso, el servicio se iniciará si los sistemas de ficheros están montados o si se está en los niveles de 2 a 5. Se detendrá en todos los niveles salvo los que se encuentran entre 2 y 5 (ambos incluidos), a saber: 0, 1 y 6. Una parte de la sintaxis retoma la de inittab, especialmente en las acciones (por ejemplo respawn).

### c. Nivel por defecto

El archivo **/etc/inittab** ya no es necesario y no se interpretará, pero hay una excepción. Si está presente, es únicamente por la línea **initdefault**. Si no existe, el nivel de ejecución predeterminado se indica en la variable **DEFAULT\_RUNLEVEL** del archivo **/etc/init/rc-sysinit.conf**. El nivel de ejecución siempre puede pasarse por parámetro al núcleo en el arranque.

### d. Compatibilidad con System V

Los scripts **rc-sysinit.conf**, **rcS.conf** y **rc.conf** siempre están para garantizar la compatibilidad con los scripts System V. El script **rc.conf** es el que va a ejecutar el archivo **/etc/init.d/rc** con el nivel de ejecución pasado por parámetro. A partir de este punto, el método es idéntico al método init System V.

### e. Comandos de control

Aunque un script init System V siempre puede controlarse con su nombre o por el comando service, estos métodos no son recomendables. upstart dispone de un cierto número de comandos que los reemplazan.

#### Iniciar un servicio

Utilice el comando start:

```
# start ufw  
ufw start/running
```

#### Recargar un servicio

Utilice el comando restart:

```
# restart ufw  
ufw start/running
```

#### Parar un servicio

Utilice el comando stop:

```
# stop ufw  
ufw stop/waiting
```

#### Estado de un servicio

Utilice el comando status:

```
# status ufw  
ufw start/running
```

#### initctl

Puede utilizar el comando initctl para controlar servicios. Con él se pueden realizar todas las acciones posibles. Recibe como primer parámetro una acción y, si procede, un segundo parámetro con el nombre del servicio asociado. Por ejemplo, puede listar los servicios:

```
# initctl list
```

```
avahi-daemon start/running, process 1410
mountall-net stop/waiting
nmbd start/running, process 2030
qemu-kvm start/running
rc stop/waiting
rpc_pipefs (idmapd) start/running
rsyslog start/running, process 1362
screen-cleanup stop/waiting
tty4 start/running, process 1511
udev start/running, process 490
upstart-udev-bridge start/running, process 486
ureadahead-other stop/waiting
apport stop/waiting
console-setup stop/waiting
...
...
```

comando también permite emitir un evento con emit. Por ejemplo, para cambiar el nivel de ejecución (lo que siempre se puede hacer con los comandos telinit, halt, reboot o shutdown):

```
# initctl emit runlevel RUNLEVEL=2
```

## f. Activación y desactivación de un servicio

No hay sustituto hasta la fecha de **chkconfig** o **update-rc.d** para los servicios upstart. Sin embargo, si la distribución mantiene cierta compatibilidad con System V y upstart sirve básicamente para controlar estos servicios, estos comandos continuarán siendo funcionales para cualquier servicio albergado en **/etc/init.d**.

En caso contrario, necesitará:

- parar manualmente el servicio,
- modificar los eventos start y stop en los servicios asociados en /etc/init.

## 2. systemd

### a. Introducción

El reemplazo de init **systemd** es fruto de una reflexión profunda sobre la ejecución de servicios, combinando todos los sistemas operativos. Su autor ha comparado especialmente init System V, upstart y el sistema de arranque de Mac OS X y ha sacado conclusiones interesantes sobre las ventajas y las limitaciones de cada uno de ellos. Se puede acceder a sus conclusiones a través de la siguiente página web: <http://0pointer.de/blog/projects/why.html>. Son bastante constructivas.

En su defensa, su autor hace algunas constataciones interesantes sobre la duración del arranque de un sistema y el número de procesos que arrancan antes de llegar a la pantalla de login. Especialmente, invita a observar cuál es el PID del primer proceso iniciado por el usuario justo después del arranque de la máquina. Ronda el 150 en Mac OS, pero es alrededor de 1800 en una distribución clásica. Por lo tanto, hay un problema. El init systemd permite acelerar el arranque de servicios, siempre basado en eventos y de forma asíncrona. Pero aún va más allá, especialmente anticipándose a las necesidades de varios servicios para acelerar su carga en el futuro, averiguar y gestionar el estado exacto de cada proceso que forme parte de un servicio y tener un control total sobre estos, sobre todo gracias a una interfaz CLI y otra gráfica. Además, aboga por el reemplazo del shell script por un lenguaje compilado.

systemd es un init reciente y por este motivo en Julio de 2011 sólo lo usaba una distribución importante, Fedora 15, en la que reemplaza a upstart. Aunque systemd es verdaderamente innovador, hay dos distribuciones que probablemente no optarán por establecerlo como predeterminado (o como mínimo, no lo harán de inmediato): Ubuntu, por una razón muy simple ya que upstart ha sido creado por Ubuntu para Ubuntu, y sobre todo Debian, por razones bastante importantes.

A pesar de que systemd utiliza mecanismos eficientes, hace invocaciones a funciones (API) únicamente presentes en el núcleo de Linux como, entre otras, el concepto de cgroups que permite agrupar procesos en un mismo grupo (cada proceso asociado a un servicio se ubica en un mismo grupo). Por ello, systemd ya no es compatible POSIX y, por lo tanto, depende de Linux. A no ser que se contribuya con aportaciones grandes y pesadas a los núcleos de otros sistemas Unix, systemd no es portable. Ahora bien, Debian no sólo está disponible para Linux. Existe una distribución Debian con un núcleo BSD y otra con el núcleo experimental HURD. Otra razón es que el autor de systemd quiere incluir el entorno de escritorio GNOME como dependencia a systemd. Cualquier distribución que use GNOME estaría obligada a usar systemd, lo que al menos tiene el mérito de haber provocado un gran debate en el mundo Linux.

## b. Unidades objetivo y servicios

En systemd, el concepto de nivel de ejecución ya sólo existe para la compatibilidad con System V. El estado deseado por el sistema después de haber ejecutado los servicios se llama la **unidad objetivo** o **target unit**. El componente básico de systemd es la **unidad** o **unit**. Existen varias: servicios, sockets, periféricos, objetivos, etc. Un objetivo es básicamente el punto de sincronización entre unidades en espera. De este modo, dispondrá, por ejemplo, de unidades objetivo correspondientes a cada nivel de ejecución, pero puede llamarles como desee.

Las definiciones de las distintas unidades se encuentran en **/lib/systemd/system**. Las unidades objetivo o los servicios que deben ser gestionados por el sistema durante el arranque se ubican en **/etc/system/system**, en forma de enlaces simbólicos o copias de los archivos anteriores.

Puede activar o definir varios objetivos al mismo tiempo y no limitarse a uno solo, único y pesado. Por ejemplo, un objetivo contendrá la activación de los servicios de red, otro los servicios de audio y un último para el inicio del entorno gráfico, etc. Un objetivo se puede agrupar en otro.

### Servicios

Los servicios acaban con el sufijo **.service**. A continuación se muestra un ejemplo con el contenido del servicio ntpd que representa sus dependencias y su comando de ejecución:

```
# cat ntpd.service
[Unit]
Description=Network Time Service
After=syslog.target ntpdate.service

[Service]
EnvironmentFile=/etc/sysconfig/ntp
ExecStart=/usr/bin/ntp -n -u ntp:ntp $OPTIONS

[Install]
WantedBy=multi-user.target
```

Una línea interesante es la de **WantedBy**. Sirve para indicar que este servicio es necesario para el objetivo multi-user.target (lo que corresponderá por ejemplo al nivel 3 del init System V). Un objetivo no define qué servicios gestiona, pero cada servicio define para qué unidad es necesario. Por lo tanto, es factible construir un árbol de dependencias.

### Objetivos

A continuación se muestra un ejemplo de la unidad **graphical.target**, que corresponde al runlevel 5:

```
# cat graphical.target
[Unit]
Description=Graphical Interface
Requires=multi-user.target
After=multi-user.target
```

El

```
Conflicts=rescue.target
AllowIsolate=yes
```

```
[Install]
Alias=default.target
```

comando siguiente devuelve la lista de servicios necesarios para este objetivo:

```
# grep -e 'WantedBy.*graphical.target' *
accounts-daemon.service:WantedBy=graphical.target
console-kit-daemon.service:WantedBy=graphical.target
firstboot-graphical.service:WantedBy=graphical.target
rtkit-daemon.service:WantedBy=graphical.target
system-setup-keyboard.service:WantedBy=graphical.target
```

Ahora bien, compruebe también que cada unidad dispone de valores **Requires** o **After**. El objetivo graphical.target requiere que multi-user.target esté parado:

```
# grep -e 'WantedBy.*multi-user.target' *
abrtd.service:WantedBy=multi-user.target
acpid.service:WantedBy=multi-user.target
atd.service:WantedBy=multi-user.target
avahi-daemon.service:WantedBy=multi-user.target
crond.service:WantedBy=multi-user.target
firstboot.service:WantedBy=multi-user.target
irqbalance.service:WantedBy=multi-user.target
mcelog.service:WantedBy=multi-user.target
NetworkManager.service:WantedBy=multi-user.target
NetworkManager-wait-online.service:WantedBy=multi-user.target
ntpdate.service:WantedBy=multi-user.target
ntpd.service:WantedBy=multi-user.target
rdisc.service:WantedBy=multi-user.target
remote-fs.service:WantedBy=multi-user.target
rsyslog.service:WantedBy=multi-user.target
smartd.service:WantedBy=multi-user.target
sssd.service:WantedBy=multi-user.target
```

cada unidad servicio puede contener valores Requires o After... Así, definiendo reglas de dependencia de forma simple (tal servicio depende de tal o cual otro y sólo debe ejecutarse después de tal o cual otro), la gestión de servicios es asíncrona.

## c. Acciones

El comando **systemctl** permite controlar la ejecución de servicios. A continuación se muestran algunos ejemplos de uso.

### Listado de unidades

Todas las unidades activas:

```
# systemctl
UNIT           LOAD  ACTIVE SUB      JOB  DESCRIPTION
...
-.mount        loaded active mounted          /
home-seb-.gvfs.mount    loaded active mounted      /home/seb/.gvfs
home.mount     loaded active mounted      /home
media.mount    loaded active mounted  Media Directory
sys-fs-f...nections.mount loaded active mounted  /sys/fs/fuse/
connections
```

```

tmp.mount                  loaded active mounted          /tmp
var-lib-...c_pipefs.mount  loaded active mounted        /var/lib/nfs/
rpc_pipefs
var-tmp.mount              loaded active mounted        /var/tmp
systemd-...plymouth.path   loaded active mounted        Forward Password
Requests to Plymouth Directory Watch
systemd-...word-wall.path  loaded active mounted        Forward Password
Requests to Wall Directory Watch
abrt-ccpp.service          loaded active mounted        LSB: Installs
log for oops messages, creates ABRT dump directories for each oops
abrtd.service              loaded active mounted        LSB: Watches system
Bug Reporting Tool
...

```

Todas las unidades:

```

# systemctl --all
...
alsa-restore.service      loaded inactive dead       Restore Sound Card
State
alsa-store.service        loaded inactive dead       Store Sound Card
State
atd.service               loaded active    running   Job spooling tools
audited.service           loaded active    running   SYSV: This starts
the Linux Auditing System Daemon, which collects security related events in
a dedicated audit log. If this
avahi.daemon.service      loaded active    running   Avahi mDNS/DNS-SD
Stack
btseed.service            loaded inactive dead       SYSV: Start up seed
clients for BitTorrent
bttrack.service           loaded inactive dead       SYSV: Start up
tracker for BitTorrent
cgconfig.service          loaded inactive dead       LSB: start and
stop the WLM configuration
cgred.service             loaded inactive dead       LSB: Start and
stop the cgroups rules engine daemon
console-...daemon.service loaded active    running   Console Manager
console-...-start.service  loaded active     exited   Console System
Startup Logging
cpuspeed.service          loaded active    exited   LSB: processor
frequency scaling support
crond.service             loaded active    running   Command Scheduler
cups.service              loaded active    running   LSB: The CUPS
scheduler
...

```

### **Parada y reinicio de unidades Cambiar el nivel de ejecución**

```

# systemctl stop cups.service
# systemctl start cups.service
# systemctl restart cups.service
# systemctl reload cups.service

```

Los niveles de ejecución se conservan por razones de compatibilidad con System V, pero son reemplazados en la práctica por unidades objetivo del mismo nombre. El nivel de ejecución puede modificarse dinámicamente detallando el nivel objetivo deseado como se describe a continuación:

```
# systemctl isolate runlevel5.target
```

Para modificar el nivel de ejecución predeterminado, debe crear un enlace simbólico llamado default.target a la unidad objetivo deseada:

```
# ln -sf /lib/systemd/system/runlevel5.target  
/etc/systemd/system/default.target
```

### **Averiguar el nivel de ejecución actual**

Como es posible que varias unidades objetivos estén activas a la vez, este concepto no tiene sentido alguno, excepto en los sistemas donde el objetivo es el nivel de ejecución y ningún otro.

El siguiente comando indica cuáles son las unidades objetivo activas:

```
# systemctl list-units -type=target  
UNIT           LOAD  ACTIVE SUB   JOB  DESCRIPTION  
basic.target    loaded active active  Basic System  
cryptsetup.target loaded active active  Encrypted Volumes  
getty.target    loaded active active  Login Prompts  
graphical.target loaded active active  Graphical Interface  
local-fs.target loaded active active  Local File System  
mail-transfer-agent.target loaded active active  Mail Transfer Agent  
multi-user.target loaded active active  Multi-User  
network.target  loaded active active  Network  
remote-fs.target loaded active active  Remote File System  
sockets.target  loaded active active  Sockets  
sound.target    loaded active active  Sound Card  
swap.target    loaded active active  Swap  
sysinit.target  loaded active active  System Initialization  
syslog.target   loaded active active  Syslog  
time-sync.target loaded active active  System Time Synchronized
```

### **d. Interfaz gráfica**

El comando **systemadm** inicia la interfaz gráfica de systemd. En Fedora, se encuentra en el paquete **systemd-gtk**. Esta interfaz permite controlar el conjunto de unidades systemd y permite realizar todas las acciones del comando **systemctl** en varias unidades.

systemd System Manager

Units Jobs

Show Only Live Units

Load State Active State Unit State Unit

loaded	active	running	rpcidmapd.service
loaded	active	running	sshd.service
loaded	active	exited	portreserve.service
loaded	active	running	rsyslog.service

Job

**Id:** sshd.service  
**Aliases:** n/a  
**Description:** LSB: Start up the OpenSSH server daemon  
**Dependencies:** requires [basic.target](#)  
conflicts [shutdown.target](#)  
wanted by [multi-user.target](#), [graphical.target](#)  
[\\_eed.service](#), [basic.target](#), [rpcbind.service](#), [bttrack.service](#), [irda.service](#), [sandbox.service](#), [dnsmasq.service](#), [netconsole.service](#)  
before [shutdown.target](#), [smolt.service](#), [multi-user.target](#), [graphical.target](#)

**Fragment Path:** n/a  
**Control Group:** name=systemd:/system/sshd.service  
**Load State:** loaded  
**Active State:** active      **Active Enter Timestamp:** lun., 18 jul. 2011 17:11:5    **Can Start/Stop:** Yes  
**Unit State:** running      **Active Exit Timestamp:** n/a      **Can Reload:** Yes

Start     Stop     Restart     Reload

*Interfaz de control de systemd.*

# Consultar el registro del sistema

## 1. dmesg

El comando **dmesg** permite recuperar los mensajes del núcleo emitidos en el momento del arranque de la máquina y los emitidos después. El registro de dmesg es circular. Después de cierto número de mensajes, los primeros desaparecen. Sin embargo, estas entradas en el registro no están perdidas, ya que el servicio syslogd (capítulo La red) las escribe en ficheros.

El administrador, ingeniero o usuario del sistema Linux suele iniciar este comando para comprobar la presencia de posibles errores. En efecto, después del boot, los mensajes siguen llegando, en particular durante la conexión en caliente de periféricos, durante la carga de algunos módulos, cuando se producen averías, durante una corrupción del sistema de ficheros, etc.

Se ha truncado voluntariamente el ejemplo siguiente a unas cincuenta líneas, ya que la salida original contiene más de 500. Las primeras muestran todo el principio de la ejecución del núcleo (información facilitada por la BIOS). En medio se muestra la detección del primer disco duro y de sus particiones. El final muestra lo que ocurre al insertar de una clave USB, después del boot, durante una utilización normal y en el momento de desconectarse.

```
# dmesg
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Linux version 3.1.5-1-ARCH (tobias@T-POWA-LX) (gcc version
4.6.2 20111125 (prerelease) (GCC) ) #1 SMP PREEMPT Sat Dec 10 14:43:09 CET 2011
[ 0.000000] Command line: root=/dev/sda2 ro vga=865
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
[ 0.000000] BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
[ 0.000000] BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
[ 0.000000] BIOS-e820: 0000000000100000 - 00000000cf7d0000 (usable)
[ 0.000000] BIOS-e820: 00000000cf7d0000 - 00000000cf7d3000 (ACPI NVS)
[ 0.000000] BIOS-e820: 00000000cf7d3000 - 00000000cf7e0000 (ACPI data)
[ 0.000000] BIOS-e820: 00000000cf7e0000 - 00000000cf800000 (reserved)
[ 0.000000] BIOS-e820: 00000000f4000000 - 00000000f8000000 (reserved)
[ 0.000000] BIOS-e820: 00000000fec00000 - 0000000100000000 (reserved)
[ 0.000000] BIOS-e820: 0000000100000000 - 000000022f800000 (usable)
[ 0.000000] NX (Execute Disable) protection: active
...
(algunas centenas de líneas)
...
[ 2.264392] scsi 1:0:0:0: Direct-Access      ATA      WDC WD10EALX-009 15.0
PQ: 0 ANSI: 5
[ 2.268102] scsi 1:0:1:0: CD-ROM           HL-DT-ST DVDRAM GH22NS50
TN03 PQ: 0 ANSI: 5
[ 2.271279] sd 1:0:0:0: [sda] 1953525168 512-byte logical blocks: (1.00 TB/931 GiB)
[ 2.271303] sd 1:0:0:0: [sda] Write Protect is off
[ 2.271305] sd 1:0:0:0: [sda] Mode Sense: 00 3a 00 00
[ 2.271315] sd 1:0:0:0: [sda] Write cache: enabled, read cache: enabled, doesn't
support DPO or FUA
[ 2.282501] sda: sda1 sda2 sda3
[ 2.285101] sd 1:0:0:0: [sda] Attached SCSI disk
...
(algunas decenas de líneas)
...
[ 4.576181] nvidia 0000:01:00.0: PCI INT A -> GSI 16 (level, low) -> IRQ 16
[ 4.576187] nvidia 0000:01:00.0: setting latency timer to 64
[ 4.576190] vgaarb: device changed decodes: PCI:0000:01:00.0,olddecodes=io+mem,
decodes=none:owns=io+mem
[ 4.576320] NVRM: loading NVIDIA UNIX x86_64 Kernel Module  290.10
```

```

Wed Nov 16 17:39:29 PST 2011
[    4.706665] snd_hda_intel 0000:01:00.1: PCI INT B -> GSI 17 (level, low) -> IRQ 17
[    4.706668] hda_intel: Disabling MSI
[    4.706736] snd_hda_intel 0000:01:00.1: setting latency timer to 64
[    4.972691] input: LITE-ON Technology USB NetVista Full Width Keyboard. as
/devices/pci0000:00/0000:00:1d.0/usb4/4-1/4-1.5/4-1.5:1.0/input/input3
[    4.972815] generic-usb 0003:04B3:3025.0001: input,hidraw0: USB HID v1.10
Keyboard [LITE-ON Technology USB NetVista Full Width Keyboard.] on
usb-0000:00:1d.0-1.5/input0
[    4.977214] input: MLK Trust Mouse 15313 as /devices/pci0000:00/0000:00:
1d.0/usb4/4-1/4-1.6/4-1.6.4/4-1.6.4:1.0/input/input4
[    4.977417] generic-usb 0003:04FC:0538.0002: input,hiddev0,hidraw1: USB HID
v1.10 Mouse [MLK Trust Mouse 15313] on usb-0000:00:1d.0-1.6.4/input0
[    4.977429] usbcore: registered new interface driver usbhid
[    4.977430] usbhid: USB HID core driver
... (y así hasta el final)

```

Para sacarle partido al resultado, lo ideal sería o bien redireccionarlo en un fichero para un análisis más en frío, o bien utilizar el comando **grep** a propósito, si sabe lo que busca.

```

# dmesg|grep CPU
[    0.000000] SMP: Allowing 8 CPUs, 4 hotplug CPUs
[    0.000000] setup_percpu: NR_CPUS:64 nr_cpumask_bits:64
nr_cpu_ids:8 nr_node_ids:1
[    0.000000] PERCPUP: Embedded 27 pages/cpu @ffff88022f400000
s80448 r8192 d21952 u262144
[    0.000000] SLUB: Genslabs=15, HWalign=64, Order=0-3, MinObjects=0,
CPUs=8, Nodes=1
[    0.000000] Verbose stalled-CPUs detection is disabled.
[    0.002591] CPU: Physical Processor ID: 0
[    0.002591] CPU: Processor Core ID: 0
[    0.002597] mce: CPU supports 9 MCE banks
[    0.002608] CPU0: Thermal monitoring enabled (TM1)
[    0.047018] CPU0: Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz stepping 07
[    0.585019] Brought up 4 CPUs
[    0.618349] Switched to NOHz mode on CPU #0
[    0.618400] Switched to NOHz mode on CPU #3
[    0.618402] Switched to NOHz mode on CPU #2
[    0.618487] Switched to NOHz mode on CPU #1

```

## 2. /var/log/messages

Sea cual sea la distribución empleada, **/var/log/messages** es el fichero central de los mensajes del sistema, provengan del núcleo o de los servicios. El contenido de este fichero, gestionado por syslog, refleja el estado global del sistema (y no únicamente del núcleo) durante su utilización. En un sistema clásico, su contenido retoma el procedente del comando **dmesg** y el de varios servicios.

Las líneas tienen una firma temporal. Sin que se produzca alguna acción especial (ver logrotate en el capítulo La red), el fichero crece en el tiempo y no es purgado. Un fichero de mensajes puede contener varios miles de líneas, sobre todo si se producen problemas!

```

# wc -l < messages
8453

```

Así, como con el comando **dmesg**, no olvide efectuar un grep para seleccionar sus líneas (o tail, head, etc.).

```

# tail -100 /var/log/messages | grep fglrx
Feb  5 09:12:22 p64p17bicb3 kernel: [fglrx] interrupt source 20008000
successfully disabled!
Feb  5 09:12:22 p64p17bicb3 kernel: [fglrx] enable ID = 0x00000000
Feb  5 09:12:22 p64p17bicb3 kernel: [fglrx] Receive disable interrupt

```

```
message with irqEnableMask: 20008000; dwIRQEnableId: 00000004
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] Maximum main memory to use
for locked dma buffers: 867 MBytes.
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] GART Table is not in FRAME_
BUFFER range
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] Reserve Block - 0 offset =
0X0 length = 0X40000
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] Reserve Block - 1 offset =
0X3ff5000 length = 0Xb000
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] interrupt source 20008000
successfully enabled
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] enable ID = 0x00000004
Feb  5 09:13:32 p64p17bicb3 kernel: [fglrx] Receive enable interrupt
message with irqEnableMask: 20008000
```

En algunas distribuciones el archivo messages puede estar vacío. Esto significará que los mensajes asociados han sido distribuidos en otros archivos por el comando syslog, del que hay una descripción más adelante en este capítulo.

# Servicios y módulos del núcleo

## 1. Presentación

El núcleo es el corazón del sistema operativo Linux. Linux, como tal, es únicamente el nombre del núcleo desarrollado originalmente por Linus Torvalds. El sistema operativo Linux se compone del núcleo y de las herramientas operativas básicas.

El núcleo de Linux es libre. Sus fuentes están disponibles. Por lo tanto, es posible volver a compilarlo para adaptarlo específicamente a sus necesidades, modificarlo y añadirle extensiones.

El núcleo de Linux forma parte de la familia de los núcleos monolíticos, es decir: que agrupan todas sus funcionalidades y componentes en un programa único. Sin embargo, desde la versión 2.0 (o más bien la versión de desarrollo 1.3 para ser más precisos) el núcleo es modular.

El núcleo se llama **kernel**. Está presente `/boot` y su nombre, por convención, comienza a menudo por `vmlinuz-X.Y.Z.p-Vtxt`.

Se obtiene la versión del núcleo con el comando **uname**.

```
$ uname -r  
2.6.38-8-generic
```

Las letras tienen un significado particular.

- **x**: versión mayor del núcleo. Entre la versión 1 y 2, el paso al funcionamiento modular ha sido determinante, así como la nueva implementación de la capa de red. La versión 3 vio la luz en verano de 2011.
- **y**: un valor par representa una rama estable del núcleo. Una versión impar representa una rama de desarrollo (¡cuidado!). Cada incremento par (0,2,4,6) representa una evolución importante del núcleo.

 La versión 2.6 no dispone todavía de una rama de desarrollo, ya que evoluciona demasiado rápido. Los desarrolladores decidieron implementar sus novedades directamente en la versión estable.

- **z**: versión menor del núcleo. Cuando un lote de modificaciones en relación con una versión anterior necesita la difusión del nuevo núcleo, entonces se incrementa esta cifra. Por ejemplo, un lote que agrupa una modificación del sistema de sonido (Alsa, que pasa de 1.0.8 a 1.0.9), del sistema de ficheros (añadido de ReiserFS 4), y así sucesivamente...
- **p**: versión corregida o intermedia presente desde la versión 2.6. Cuando el núcleo necesita una actualización menor (corrección de uno o dos bugs, etc.) pero pocas funcionalidades nuevas, se incrementa este valor.
- **v**: como para los paquetes, versión propia del editor de la distribución.
- **txt**: a veces se añade un texto para dar precisiones sobre el núcleo. Por ejemplo, `smp` indica un núcleo multiprocesador.

La versión 3.0 salió en verano de 2011. Aunque sea una etapa muy importante, la numeración en la versión 3.0 sólo es una decisión que se podría catalogar como política. Entre las versiones del núcleo 2.6.0 y 2.6.39 han transcurrido bastantes años y ha habido un gran número de modificaciones que han provocado que entre ambas versiones haya poco en común. Como el núcleo 2.6 no ha tenido una rama de desarrollo, la diferencia entre las versiones 2.6.39 y 3.0 no serán tan importantes que como las versiones anteriores. La versión 3.0

se presenta de este modo por parte de Linus Torvalds:

*"Entonces, ¿cuáles son los cambios más importantes? NINGUNO. Absolutamente ninguno. Por supuesto, tenemos las habituales modificaciones de dos o tres drivers y una gran cantidad de correcciones, pero la versión 3.0 no es más que un simple cambio de numeración, no como con KDE-4 o GNOME-3. La compatibilidad no se ha perdido, ya que no hay nuevas funcionalidades que asusten ni nada parecido. A partir de ahora, se publican nuevas versiones basándose en las fechas y no en las novedades funcionales, tal y como se ha estado haciendo en mucho tiempo. Si quiere una excusa para esta reenumeración, digamos que ha sido por los 20 años de Linux."*

Por lo tanto, el gran cambio sólo concierne al nuevo modo de numeración.

## 2. uname

El comando **uname** (unix name) permite obtener toda la información correspondiente a la versión de Unix (de Linux aquí), de manera precisa y completa.

Parámetro	Resultado
-m (máquina)	Tipo de hardware de la máquina.
-n (nodename)	Nombre de la máquina.
-r (release)	Versión (número) del núcleo.
-s (system name)	Nombre del sistema operativo. Por defecto.
-p (processor)	Tipo de procesador.
-i	Plataforma física.
-v (version)	Versión del sistema.
-a (all)	Toda la información.

```
$ uname
Linux
$ uname -m
X86_64
$ uname -n
slyserver
$ uname -r
2.6.38-8-generic
$ uname -s
Linux
$ uname -p
$ uname -i
X86_64
$ uname -o
GNU/Linux
$ uname -v
#42-Ubuntu SMP Mon Apr 11 03:31:24 UTC 2011
$ uname -a
Linux slyserver 2.6.38-8-generic #42-Ubuntu SMP Mon Apr 11 03:31:24 UTC 2011
x86_64 intel x86_64 GNU/Linux
```

### 3. Gestión de los módulos

Los componentes básicos (gestor de tareas, gestión de la memoria, de los procesos, API, etc.) están siempre presentes dentro de un programa único. Pero algunos drivers de periférico, sistemas de ficheros, extensiones, protocolos de redes, etc., pueden estar presentes en forma de módulos. Los módulos se comunican con el núcleo a través de un API común. Se ejecutan en el espacio del núcleo. Se pueden configurar. Se pueden cargar y descargar a petición y así evitar un reinicio de la máquina. El añadido de un nuevo módulo (desde sus fuentes por ejemplo) no necesita reinicio.

Los módulos están presentes en `/lib/modules/$(uname -r)`.

```
# cd /lib/modules/$(uname -r)
# pwd
/lib/modules/2.6.38-8-generic
```

Los módulos tienen un nombre que termina por «ko» para kernel object. La terminación de origen era «o» para los núcleos 2.0 a 2.4. Se trata efectivamente de esto: ficheros de objetos relacionados (linked) de manera dinámica al núcleo, que proponen así una API adicional.

```
# cd /lib/modules/$(uname -r)/kernel/fs/vfat
# file vfat.ko
vfat.ko: ELF 64-bit LSB relocatable, x86-64, version 1 (SYSV),
not stripped
```

La palabra «relocatable» indica que usted está ante un fichero de objeto.

#### a. **lsmod**

El comando **lsmod** lista los módulos actualmente cargados, con sus posibles dependencias.

```
seb@debian:~$ lsmod
Module           Size  Used by
cpufreq_userspace    1992  0
cpufreq_powersave     902   0
cpufreq_stats        2740   0
cpufreq_conservative 5162   0
vboxvideo            1765   1
drm                142279  2 vboxvideo
ppdev               5030   0
lp                  7462   0
sco                 7225   2
bridge              39646  0
stp                  1440   1 bridge
bnep                 9427   2
rfcomm               29629  0
l2cap                24752  4 bnep,rfcomm
bluetooth            41827  6 sco,bnep,rfcomm,l2cap
rfkill                13044  2 bluetooth
nfsd                 254782 11
lockd                 57619  1 nfsd
nfs_acl                2031  1 nfsd
auth_rpcgss            33508  1 nfsd
...
```

La primera columna indica el nombre del módulo cargado. Su nombre refleja a menudo para qué sirve. La segunda columna da el tamaño del módulo. La tercera columna suministra un contador de utilización (cuántos componentes del sistema acceden a los módulos). La última columna proporciona la lista de los módulos que utilizan (o, por lo tanto, dependen) del primero.

En el ejemplo anterior el módulo vfat utiliza el módulo fat.

**lsmod** sólo vuelve a dar forma al contenido del fichero virtual **/proc/modules**.

```
seb@debian:~$ cat /proc/modules
cpufreq_userspace 1992 0 - Live 0xfffffffffa004e000
cpufreq_powersave 902 0 - Live 0xfffffffffa00ac000
cpufreq_stats 2740 0 - Live 0xfffffffffa0082000
cpufreq_conservative 5162 0 - Live 0xfffffffffa004a000
vboxvideo 1765 1 - Live 0xfffffffffa001a000
drm 142279 2 vboxvideo, Live 0xfffffffffa03ed000
ppdev 5030 0 - Live 0xfffffffffa03e9000
lp 7462 0 - Live 0xfffffffffa03e1000
sco 7225 2 - Live 0xfffffffffa03d9000
bridge 39646 0 - Live 0xfffffffffa03c6000
stp 1440 1 bridge, Live 0xfffffffffa03c0000
bnep 9427 2 - Live 0xfffffffffa03b7000
rfcomm 29629 0 - Live 0xfffffffffa03a8000
l2cap 24752 4 bnep,rfcomm, Live 0xfffffffffa039b000
bluetooth 41827 6 sco,bnep,rfcomm,l2cap, Live 0xfffffffffa0386000
rfkill 13044 2 bluetooth, Live 0xfffffffffa037b000
...
...
```

## b. depmod

El comando **depmod** actualiza el árbol de las dependencias entre los módulos modificando el fichero **modules.dep**.

El fichero **/lib/modules/\$(uname -r)/modules.dep** contiene dos columnas. La primera columna es la ruta del módulo, la segunda es la lista de las dependencias: los módulos que se deben cargar también para que el primero funcione. Aquí, tenemos el ejemplo de la línea que corresponde al módulo **vfat**:

```
$ grep vfat modules.dep
kernel/fs/fat/vfat.ko.gz: kernel/fs/fat/fat.ko.gz
```

módulo **vfat.ko** depende del módulo **fat.ko**. Por lo tanto se debe cargar el módulo **fat.ko** primero para que **vfat.ko** funcione.

El uso más corriente de **depmod** incluye el parámetro **-a**, que vuelve a construir las dependencias de todos los módulos que corresponden al núcleo actual. Se ejecuta esta acción a cada inicio del sistema, pero si usted compila o instala nuevos módulos, debe reiniciar manualmente este comando para tener en cuenta las nuevas dependencias.

```
# depmod -a
```

## c. modinfo

El comando **modinfo** provee toda la información necesaria sobre un módulo:

- el nombre del fichero correspondiente,
- una descripción del módulo,
- su autor,
- su licencia,
- sus dependencias,
- sus parámetros,

- sus alias.

Los módulos no proporcionan toda esta información. Algunos módulos no tienen parámetros. En el primer ejemplo, el módulo `vfat` no tiene parámetros. Éstos se instalan como opciones de montaje.

```
$ modinfo vfat
filename:      /lib/modules/3.1.5-1-ARCH/kernel/fs/fat/vfat.ko.gz
author:        Gordon Chaffee
description:   VFAT filesystem support
license:       GPL
depends:      fat
vermagic:     3.1.5-1-ARCH SMP preempt mod_unload modversions
```

En este segundo ejemplo, el módulo dispone de parámetros. Este módulo permite utilizar una webcam que contiene un chip compatible con el módulo `uvccvideo`. Los parámetros están truncados de manera voluntaria.

```
$ modinfo uvccvideo
filename:      /lib/modules/3.1.5-1-ARCH/kernel/drivers/media/video/uvc/
uvccvideo.ko.gz
version:       1.1.1
license:       GPL
description:  USB Video Class driver
author:        Laurent Pinchart <laurent.pinchart@ideasonboard.com>
srcversion:   DCA7C031F668695E71EF3E4
alias:        usb:v*p*d*dc*dsc*dp*ic0Eisc01ip00*
alias:        usb:v1C4Fp3000d*dc*dsc*dp*ic0Eisc01ip00*
alias:        usb:v1B3Bp2951d*dc*dsc*dp*ic0Eisc01ip00*
alias:        usb:v19ABp1000d00*dc*dsc*dp*ic0Eisc01ip00*
alias:        usb:v19ABp1000d01[0-1]*dc*dsc*dp*ic0Eisc01ip00*
alias:        usb:v19ABp1000d012[0-6]dc*dsc*dp*ic0Eisc01ip00*
...
depends:     videodev,usbcore,media
vermagic:    3.1.5-1-ARCH SMP preempt mod_unload modversions
parm:        clock:Video buffers timestamp clock
parm:        nodrop:Don't drop incomplete frames (uint)
parm:        quirks:Forced device quirks (uint)
parm:        trace:Trace level bitmask (uint)
parm:        timeout:Streaming control requests timeout (uint)
```

Un mismo módulo puede gestionar varios tipo de componentes físicos. Existen varias webcams que disponen de un chip ov511 o similar que se pueden gestionar con el módulo `uvccvideo`. Para que el núcleo sepa qué módulo cargar durante la detección de la webcam, o para que el módulo sepa qué tipo de componente físico debe gestionar durante su carga, dispone de alias de hardware, `usb`, `pci`, `scsi`, etc., que permiten reconocer los periféricos que debe gestionar.

Se pasan los parámetros al módulo mediante los comandos `insmod` o `modprobe`, usando el fichero `/etc/modprobe.conf`.

## d. insmod

El comando `insmod` carga un módulo dado sin gestionar las dependencias. Coge como parámetro un nombre de módulo con su ruta. Le corresponde a usted gestionar el orden de carga de los módulos para evitar errores relacionados con símbolos no resueltos causados por un problema de dependencia.

En el ejemplo siguiente, como los módulos `fat` y `vfat` no están cargados, se hace un intento para cargar el módulo `vfat.ko` solo. Se produce un error, ya que este módulo depende sólo de `fat.ko`. El retorno de `dedmesg` es elocuente al respecto.

```
# lsmod|grep fat
```

```
# ls
vfat.ko
# insmod vfat.ko
insmod: error inserting 'vfat.ko': -1 Unknown symbol in module
# dmesg | tail -20
...
vfat: Unknown symbol fat_dir_empty
vfat: Unknown symbol fat_fs_panic
vfat: Unknown symbol fat_get_dotdot_entry
vfat: Unknown symbol fat_free_clusters
vfat: Unknown symbol fat_scan
vfat: Unknown symbol fat_date_unix2dos
vfat: Unknown symbol fat_search_long
vfat: Unknown symbol fat_getattr
vfat: Unknown symbol fat_attach
vfat: Unknown symbol fat_build_inode
vfat: Unknown symbol fat_fill_super
vfat: Unknown symbol fat_alloc_new_dir
vfat: Unknown symbol fat_notify_change
vfat: Unknown symbol fat_remove_entries
vfat: Unknown symbol fat_add_entries
vfat: Unknown symbol fat_sync_inode
vfat: Unknown symbol fat_detach
```

En este segundo ejemplo, primero se carga el módulo fat, luego el módulo vfat. No hay error, ya que todas las dependencias están presentes.

```
# cd ../fat
# ls
fat.ko
# insmod fat.ko
# cd ../vfat
# insmod vfat.ko
# lsmod|grep fat
vfat              16128  0
fat                52636  1 vfat
```

Debería considerar el parámetro `-k`, que permite la limpieza automática de los módulos. Esta opción solicita al sistema que descargue automáticamente un módulo (contador a cero) si ya no se utiliza. Esto permite ganar algunos recursos y obtener un sistema más limpio.

Para transmitir parámetros al módulo, indíquelos a continuación del comando. Por ejemplo, para el módulo uvcvideo, desea que el driver muestre más información:

```
# insmod uvcvideo.ko trace=5
```

## e. rmmod

El comando **rmmod** descarga el módulo dado. Se trata del contrario de **insmod** y, al igual que éste, rmmod no gestiona las dependencias:

- No es posible descargar un módulo en curso de utilización.
- No es posible descargar un módulo si otro módulo lo utiliza, incluso si este último no es utilizado (problema de dependencia).

En este ejemplo, un dispositivo USB que contiene un sistema de ficheros vfat ha sido conectado y montado. Un primer intento de suprimir vfat fracasa.

```
# mount | grep vfat
```

```
/dev/sdb1 on /media/disk type vfat ...
# rmmod vfat
ERROR: Module vfat is in use
```

En este segundo ejemplo, el dispositivo está desconectado. Los módulos fat y vfat se han vuelto inútiles. Se intenta suprimir el módulo fat. El sistema devuelve un error relacionado con las dependencias.

```
# rmmod fat
ERROR: Module fat is in use by vfat
```

En este último ejemplo, se descarga el módulo vfat, luego el módulo fat, en este orden.

```
# rmmod vfat
# rmmod fat
```

## f. modprobe

El comando **modprobe** carga el módulo dado, así como todas sus dependencias y parámetros contenidos en `/etc/modprobe.conf`. El parámetro `-r` permite descargar un módulo y los que dependen de él (si no están en uso).

Cargar el módulo vfat mediante **modprobe** significa en la práctica cargar también el módulo fat.

```
# lsmod|grep fat
# modprobe vfat
# lsmod|grep fat
vfat          16128  0
fat           52636  1 vfat
```

Ahora, como ve que sólo vfat utiliza fat (contador a 1), pero que nada utiliza vfat (contador a cero), puede intentar descargar vfat y los módulos de los cuales depende, si no están en uso.

```
# modprobe -r vfat
# lsmod|grep fat
```

 Puede pasar opciones al módulo; en ese caso, debe comprobar si una línea «install» que corresponde al módulo existe en el fichero `modprobe.conf` y modificarla para que acepte los parámetros, o utilizar una línea «opciones».

## g. modprobe.conf y modprobe.d

 ¡Atención! El uso del archivo `modprobe.conf` ha caído en desuso en las versiones recientes. Cuando está presente, se muestra un aviso. Concretamente, este archivo se ignorará completamente. La configuración se traslada a los archivos independientes albergados en `/etc/modprobe.d`.

La configuración de los módulos se encuentra en `/etc/modprobe.conf` y en los archivos albergados en `/etc/modprobe.d/*.conf`. Aunque las distribuciones actuales han dejado de utilizar `modprobe.conf`, sigue siendo importante, ya que todavía se utiliza en distribuciones empresariales, especialmente en las distribuciones SLES, RHEL y las antiguas Debian. En estos archivos la sintaxis es idéntica. Puede definir alias de módulos (muy práctico para las tarjetas de red), pasar opciones a los módulos así como añadir acciones en la carga y descarga de un módulo.

### Alias y opciones

Pongamos un ejemplo: usted tiene dos tarjetas de red. El driver de la primera tarjeta se encuentra en el módulo **e1000**. Quiere que se cargue utilizando el nombre **eth0**. El driver de la segunda tarjeta está en el módulo **airo** y desea cargarlo utilizando el nombre **eth1**. También va a especificar parámetros al módulo **uvccvideo**, cuya información ha recuperado con **modinfo**.

```
# cat /etc/modprobe.d/network-interfaces.conf  
...  
alias eth0 e1000  
alias eth1 airo  
# cat /etc/modprobe.d/webcam.conf  
options uvccvideo led=2 compress=1  
...
```

Luego carga los módulos por nombre o por alias. En todos los casos, se tienen en cuenta los parámetros automáticamente.

```
# modprobe eth0  
# modprobe eth1  
# modprobe uvccvideo
```

Se pueden utilizar las opciones de módulos con los alias.

```
alias webcam uvccvideo  
options webcam led=2 compress=1  
...
```

### install y remove

Los comandos **install** y **remove** del fichero **modprobe.conf** son los más potentes. Durante la carga de un módulo, si **modprobe** encuentra un comando **install** asociado, no carga este módulo, sino que ejecuta los comandos indicados. El comando puede ser lo que usted desee, como por ejemplo la carga de un módulo más adaptado, la instalación de una configuración específica, la ejecución de un script, etc.

El ejemplo siguiente intenta cargar el módulo **uvccvideo\_new** si existe, y si no, conmuta a **uvccvideo** cuando usted invoca **modprobe webcam**.

```
install webcam /sbin/modprobe uvccvideo_new || /sbin/modprobe uvccvideo
```

**modprobe** puede coger un parámetro práctico que le permita ignorar las líneas **install** de **modprobe.conf**:--**ignore-install**. El ejemplo siguiente carga el módulo **ahci**, luego el módulo **ata\_piix** cuando usted intenta cargar este último. Sin el parámetro, **modprobe** se ejecutaría en bucle, leyendo e interpretando la línea **install** a cada intento de carga de **ata\_piix**.

```
install ata_piix /sbin/modprobe ahci 2>&1 |:; /sbin/modprobe --  
ignore-install ata_piix
```

comando **remove** hace lo mismo, pero en el momento de descargar el módulo con **modprobe -r**.

### CMDLINE\_OPTS

Si una línea **install** está presente en **modprobe.conf** e intenta cargar el módulo correspondiente con **modprobe** y unos parámetros, no se tendrán en cuenta estos últimos excepto si usted ha añadido detrás del módulo la cadena **\$CMDLINE\_OPTS**.

```
install webcam /sbin/modprobe ov511_new $CMDLINE_OPTS ||  
/sbin/modprobe ov511 $CMDLINE_OPTS
```

## 4. Carga de los módulos al inicio

### a. initrd

Algunos módulos pueden ser necesarios en el inicio de la máquina, en particular para montar un sistema de ficheros. ¿Cómo montar la partición raíz en ext3 cuando el módulo que gestiona este tipo de sistema de ficheros está en esta partición (y no en el núcleo)? Se colocan estos módulos en forma de imagen de disco virtual o **initrd** (initial ramdisk). Se cargan estos ficheros comprimidos al inicio en memoria y son ramdisks. Contienen instrucciones y módulos que se cargan al inicio.

```
# ls -l /boot/initrd.img-*
-rw-r--r-- 1 root root 19011977 2011-05-02 21:04 /boot/
initrd.img-2.6.35-28-generic
-rw-r--r-- 1 root root 19011977 2011-07-18 18:43 /boot/
initrd.img-2.6.38-10-generic
-rw-r--r-- 1 root root 19011977 2011-07-05 20:46 /boot/
initrd.img-2.6.38-8-generic
```

Es posible extraer el contenido de **initrd** e incluso modificarlo y volver a construirlo para adaptarlo. Este fichero es una carpeta cpio comprimida con gzip.

```
# zcat /boot/initrd.img-2.6.38-10-generic |cpio -id --no-absolute-filenames
103092 blocks
# ls -l
total 40
drwxr-xr-x 2 root root 4096 2011-07-18 22:07 bin
drwxr-xr-x 3 root root 4096 2011-07-18 22:07 conf
drwxr-xr-x 8 root root 4096 2011-07-18 22:07 etc
-rwxr-xr-x 1 root root 4096 2011-07-18 22:07 init
drwxr-xr-x 8 root root 4096 2011-07-18 22:07 lib
drwxr-xr-x 2 root root 4096 2011-07-18 22:07 lib64
drwxr-xr-x 2 root root 4096 2011-07-18 22:07 sbin
drwxr-xr-x 10 root root 4096 2011-07-18 22:07 scripts
drwxr-xr-x 4 root root 4096 2011-07-18 22:07 usr
```

Cuando se carga y monta **initrd**, el núcleo intenta ejecutar el script **init** presente en la raíz del seudosistema de ficheros. Es el encargado de cargar los módulos necesarios y establecer la primera configuración básica completa, antes de que el núcleo monte el sistema de fichero raíz y ejecute **/sbin/init**. En la práctica, este fichero (o uno invocado por éste) es el encargado de montar el sistema de ficheros raíz y de pasarle el control a **/sbin/init**.

 En algunas distribuciones, en particular las antiguas, se invoca a un script llamado **/sbin/init**. Se ha suprimido esto para evitar confundir el script **/sbin/init** de **initrd** con el «verdadero» **init** del sistema. Si un fichero **linuxrc** se encuentra en la raíz del **initrd**, se ejecuta. En caso de duda en cuanto al nombre del script iniciado, se puede ver en las fuentes del núcleo **init/main.c**, variable **ram\_execute\_command**.

```
$ cd /usr/src/linux
$ grep -n ram_execute_command init/main.c
...
865:           ramdisk_execute_command = "/init";
...
```

comando **mkinitrd** permite volver a construir un fichero **initrd** estándar, pero a menudo es posible, en función de la distribución, facilitarle una lista de módulos para colocarlos en él, y así cargarlos:

- En Red Hat, se puede utilizar varias veces la opción **--preload=module**, que especifica cuáles son los módulos que se deben cargar.
- En openSUSE, le corresponde a la opción **-m "module1 module2 etc."** especificar esta lista. Veremos el método con la utilización de las variables de sysconfig.
- En Debian, hay que colocar los nombres de los módulos en el fichero **/etc/mkinitrd/modules**.

Tenemos el resultado siguiente para un comando **mkinitrd** en openSUSE. La línea «Kernel Modules» especifica los módulos cargados por **initrd**.

```
# mkinitrd

Kernel image: /boot/vmlinuz-2.6.22.17-0.1-default
Initrd image: /boot/initrd-2.6.22.17-0.1-default
Root device: /dev/disk/by-id/scsi-SATA_ST380011A_5JVTH798-part6
(/dev/sda6) (mounted on / as ext3)
Resume device: /dev/sda5
Kernel Modules: processor thermal scsi_mod libata sata_sil
pata_atiixp fan jbd mbcache ext3 edd sd_mod usbcore ohci-hcd uhci-hcd
ehci-hcd ff-memless hid usbhid
Features: block usb resume.userspace resume.kernel
Bootsplash: SuSE (1280x1024)
17588 blocks
```

Un initrd construido correctamente (o más bien un comando **mkinitrd** correcto) recupera también la salida del comando **modprobe** y el fichero de configuración **modprobe.conf**, que puede (y debe) contener los parámetros de los módulos si necesario.

Las últimas versiones de Debian y Ubuntu ya no utilizan **mkinitrd**, sino **mkinitramfs**. El principio es el mismo. **initramfs** dispone de su propia configuración situada en **/etc/initramfs-tools/\***. Los módulos que se cargarán en el arranque se especifican en **/etc/initramfs-tools/modules**. Para crear un initrd para el núcleo actual, utilice el comando siguiente:

```
# mkinitramfs -o /tmp/initramfs-$(uname -r)
# ls -l /tmp/initramfs-2.6.38-8-generic
-rw-r--r-- 1 root root 19812003 2011-07-18 22:16 /tmp/
initramfs-2.6.38-8-generic
```

## b. Red Hat /etc/rc.modules

El método preferido en Red Hat consiste en cargar los módulos desde el initrd. Sin embargo, tiene la posibilidad crear un script **/etc/rc.modules** que contendrá los comandos necesarios para la carga de los módulos, principalmente con **modprobe**, como vimos más arriba. Se ejecuta este fichero con **rc.sysinit** en el momento del inicio del sistema, vía **inittab**. Debe ser ejecutable.

## c. openSUSE: /etc/sysconfig/kernel

La distribución openSUSE coloca su configuración en un directorio **/etc/sysconfig**. El fichero **/etc/sysconfig/kernel** contiene unos elementos de configuración del núcleo, pero, sobre todo, dos variables encargadas de especificar los módulos que se deben cargar:

- **INITRD\_MODULES** es la lista de los módulos cargados por el initrd.
- **MODULES\_LOADED\_ON\_BOOT** es la lista de módulos cargados por init (por lo tanto, después de initrd).

Verá una diferencia entre la lista de los módulos indicada para el initrd y la lista real cargada durante la creación de éste. El comando **mkinitrd** de openSUSE permite especificar funcionalidades preestablecidas

durante la creación del initrd (opción **-f "feature1 feature2 etc"**). Por ejemplo, la funcionalidad «usb» va a cargar todos los módulos relacionados con el soporte USB, lo que resulta vital si dispone de un teclado USB o si inicia el sistema desde un dispositivo USB o un disco externo.

#### d. Debian: /etc/modules

En Debian, basta con añadir los nombres de los módulos que se deben cargar en el fichero **/etc/modules**. Después del nombre de los módulos, puede indicar parámetros. Para cada línea se ejecuta un modprobe. Puede añadir un comentario después de una **#**. El fichero **/etc/init.d/modutils** carga los módulos en el momento del inicio vía init System V, o **/etc/init/module-init-tools.conf** vía upstart.

```
# /etc/modules: kernel modules to load at boot time.
#
# This file should contain the names of kernel modules that are
# to be loaded at boot time, one per line. Comments begin with
# a "#", and everything on the line after them are ignored.

ide-cd
ide-detect
ov511
```

### 5. Parámetros dinámicos

#### a. /proc y /sys

**/proc** y **/sys** son sistemas de ficheros virtuales que contienen información relativa al núcleo en curso de ejecución. La versión 2.4 del núcleo sólo conoce **/proc**, donde se agrupa toda la información. La versión 2.6 del núcleo ha modificado la función de **/proc** para delegar una parte de sus tareas a **/sys**.

Al tratarse de sistemas de ficheros virtuales, no ocupan espacio ni en la memoria, ni en cualquier disco. Uno no debe dejarse engañar por el tamaño de los seudoficheros que contiene. ¡No intente suprimir **/proc/kcore** para ganar espacio! Se puede leer y visualizar directamente todos estos ficheros (o casi).

Los ficheros de **/proc** le proporcionarán mucha información sobre el sistema:

- **interrupts**: los parámetros IRQ.
- **cputime**: detalles sobre sus procesadores.
- **dma**: los parámetros DMA.
- **ioports**: los puertos de memoria de E/S.
- **devices**: los periféricos presentes.
- **meminfo**: el estado global de la memoria.
- **loadavg**: la carga del sistema.
- **uptime**: tiempo transcurrido desde el arranque.
- **version**: detalles de la versión de Linux.
- **modules**: idéntico al resultado de **lsmod**.
- **swaps**: lista y estado de las particiones de intercambio.

- **partitions**: lista y estado de las particiones conocidas del sistema.
- **mounts**: montaje de los sistemas de ficheros.
- **pci**: detalles del bus PCI.

```
$ cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 42
model name     : Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz
stepping        : 7
cpu MHz         : 3292.323
cache size      : 6144 KB
physical id    : 0
siblings        : 4
core id         : 0
cpu cores       : 4
apicid          : 0
initial apicid : 0
fpu             : yes
fpu_exception   : yes
cpuid level    : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca
cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx
rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology
nonstop_tsc aperfmpf perf pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2
ssse3 cx16 xtpr pdcm sse4_1 sse4_2 x2apic popcnt aes xsave avx lahf_lm ida arat
epb xsaveopt pln pts dts tpr_shadow vnmi flexpriority ept vpid
bogomips        : 6587.00
clflush size    : 64
cache_alignment : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:

$ cat /proc/version
Linux version 3.1.5-1-ARCH (tobias@T-POWA-LX) (gcc version 4.6.2
20111125 (prerelease) (GCC) ) #1 SMP PREEMPT Sat Dec 10 14:43:09 CET 2011
```

/proc contiene subdirectorios que agrupan información por tema.

- **/proc/scsi**: información sobre el bus SCSI.
- **/proc/ide**: información sobre el bus IDE.
- **/proc/net**: información sobre la red.
- **/proc/sys**: parámetros y configuración dinámica del núcleo.
- **/proc/<PID>**: información sobre el proceso PID.

Algunas entradas de los sistemas de ficheros **/proc/sys** y **/sys** son diferentes de los demás, ya que se puede modificar su contenido y el núcleo tiene en cuenta las modificaciones directamente, sin que sea preciso iniciar de nuevo la máquina.

Por ejemplo, veamos cómo activar el forwarding IP y pasar el número de manejadores de ficheros de 8192 a 16384.

```
# echo "1" > /proc/sys/net/ipv4/ip_forward
# echo "16384" > /proc/sys/fs/file-max
```

---

He aquí otro ejemplo relacionado con el capítulo anterior y la gestión de la memoria alta y baja. En los antiguos núcleos (por ejemplo 2.6.18) se podía modificar el ratio de protección de la memoria baja lowmem modificando **lower\_zone\_protection**. Se indica que el sistema debe intentar proteger lo mejor posible una zona de 300 MB de la memoria lowmem.

```
# echo 300>/proc/sys/vm/lower_zone_protection
```

Los núcleos recientes usan **lowmem\_reserve\_ratio**. Los tres valores se expresan en ratio 1/n de páginas de memoria que se deben proteger por zona de memoria. Para reservar más páginas, basta con reducir los valores:

```
# echo "128 128 32" >/proc/sys/vm/lowmem_reserve_ratio
```

## b. sysctlv

Los valores modificados en caliente no se registran. En caso de un nuevo inicio, hay que empezar de nuevo. El fichero **rc.sysinit** llama al comando **sysctl**, que actúa sobre estos parámetros. Para que los valores se queden permanentes (nueva instalación a cada inicio), hay que modificar el fichero/**/etc/sysctl.conf**. Usted puede buscar las modificaciones manualmente.

```
# sysctl -e -p /etc/sysctl.conf
# sysctl -a
...
dev.raid.speed_limit_max = 100000
dev.raid.speed_limit_min = 100
net.token-ring.rif_timeout = 60000
net.ipv4.conf.eth1.arp_filter = 0
net.ipv4.conf.eth1.tag = 0
net.ipv4.conf.eth1.log_martians = 0
net.ipv4.conf.eth1.bootp_relay = 0
net.ipv4.conf.eth1.proxy_arp = 0
net.ipv4.conf.eth1.accept_source_route = 1
net.ipv4.conf.eth1.send_redirects = 1
net.ipv4.conf.eth1.rp_filter = 1
net.ipv4.conf.eth1.shared_media = 1
net.ipv4.conf.eth1.secure_redirects = 1
...
```

# Compilar un núcleo

## 1. Obtener las fuentes

### a. Fuentes oficiales

Las fuentes oficiales del núcleo están disponibles desde el sitio kernel.org. Llevan el nombre de «vanilla». Un núcleo (o kernel) vanilla es un núcleo bruto, sin añadido de parches, procedente directamente de los desarrolladores que colaboran en el núcleo, y no ha sido adaptado aún a ninguna distribución.

Una vez se haga con las fuentes, debe configurar, compilar e instalar un núcleo, y si es preciso crear un initrd antes de poder utilizarlo.

El núcleo se proporciona en forma de carpeta comprimida que usted debe abrir con las herramientas adaptadas.

```
# ls
linux-3.0-rc7.tar.xz
# tar xvJf linux-3.0-rc7.tar.xz | more
linux-3.0-rc7/
linux-3.0-rc7/.gitignore
linux-3.0-rc7/.mailmap
linux-3.0-rc7/COPYING
linux-3.0-rc7/CREDITS
linux-3.0-rc7/Documentation/
linux-3.0-rc7/Documentation/.gitignore
linux-3.0-rc7/Documentation/00-INDEX
linux-3.0-rc7/Documentation/ABI/
linux-3.0-rc7/Documentation/ABI/README
linux-3.0-rc7/Documentation/ABI/obsolete/
linux-3.0-rc7/Documentation/ABI/obsolete/proc-pid-oom_adj
linux-3.0-rc7/Documentation/ABI/obsolete/sysfs-bus-usb
linux-3.0-rc7/Documentation/ABI/obsolete/sysfs-class-rfkill
linux-3.0-rc7/Documentation/ABI/obsolete/sysfs-driver-hid-roccat-koneplus
linux-3.0-rc7/Documentation/ABI/removed/
...
...
```

Se deben ubicar las fuentes del núcleo en `/usr/src/linux`. Si en el sistema de ficheros se encuentran varias versiones de los ficheros fuente del núcleo, Linux debe ser el vínculo simbólico hacia la fuente del núcleo actual.

```
# ls -l
total 74944
lrwxrwxrwx 1 root src      13 2011-07-18 22:34 linux -> linux-3.0-rc7
drwxrwxr-x 23 root root    4096 2011-07-12 01:51 linux-3.0-rc7
```

### b. Fuentes de la distribución

Las distribuciones aportan núcleros a menudo parcheados. Estas modificaciones revisten varios aspectos: añadido de drivers, backports (actualizaciones retroactivas, añadido de funcionalidades procedentes de una versión ulterior) de núcleos más recientes, actualizaciones de seguridad, añadido de funcionalidades, etc. Se aplican a un núcleo vanilla, y a menudo en un orden preciso.

Compilar un núcleo no es una nadería. Si ha firmado un contrato de asistencia con el editor de la distribución (por ejemplo, para una versión Servidor de Red Hat) y usted encuentra problemas con el núcleo por defecto, debería pensar en:

- comprobar si una actualización oficial corrige su problema,
- subir su problema al foro de asistencia del editor,

antes de proceder a instalar un núcleo recompilado por usted mismo.

Si a pesar de todo usted desea recompilar el núcleo, el trámite es el mismo que para las fuentes oficiales, sólo que debe instalar el paquete de las fuentes, lo que debería tener como efecto la instalación al mismo tiempo de todas las herramientas necesarias para la compilación.

El paquete se llama kernel-sources en openSUSE y Red Hat. En cuanto a este último, diríjase al sitio Red Hat, ya que el paquete **kernel-sources** no es necesario para compilar nuevos módulos (hace falta el paquete kernel-devel). No es muy fácil obtenerlos. En Debian, el paquete se llama **linux-source-2.x.y** (x e y representan la versión del núcleo). Debe instalar estos paquetes con las herramientas adaptadas a su distribución.

## 2. Las herramientas necesarias

Para compilar el núcleo de Linux, es necesario disponer de determinadas herramientas:

- el compilador C;
- las librerías de desarrollo C estándar;
- la librería ncurses (para menuconfig);
- la librería qt3 (para xconfig);
- las herramientas Make;
- los modutils;
- mkinitrd;
- concentración;
- nervios de acero;
- paciencia;

Estas herramientas son proporcionadas de forma automática por la distribución, salvo las tres últimas...

## 3. Configuración

### a. El .config

Antes de lanzar la compilación del núcleo, debe seleccionar las opciones, funcionalidades y drivers que desea conservar o no. A continuación le mostraremos varias formas de hacer esto. Luego se guarda esta configuración en un fichero **/usr/src/linux/.config**. Este fichero contiene un gran número de variables: cada una de ellas corresponde a una opción del núcleo, y pueden coger tres valores:

- **y**: la funcionalidad está presente e integrada dentro del núcleo monolítico, o, si depende de un módulo, integrada dentro de este módulo;
- **m**: se compilará la funcionalidad en forma de módulo;
- **n**: la funcionalidad está ausente.

En el último caso, este valor está presente pocas veces. Basta con que la opción esté ausente del fichero para que no se active durante la compilación. En este caso, se comenta la línea correspondiente simplemente con una # delante.

El ejemplo siguiente muestra que la opción MCORE2 (optimización para los Intel Core2) no está activa.

```
$ grep "^#" .config | grep -i core2  
# CONFIG_MCORE2 is not set
```

Ejemplo siguiente muestra las líneas de configuración asociadas a las funcionalidades del sistema de ficheros ext3. Se compila el soporte del sistema de fichero ext3 en forma de módulo (primera línea). Las funcionalidades adicionales de ext3 (atributos extendidos, ACL, seguridad, atributos nfs4) dependen de la primera línea. Por lo tanto, estarán presentes dentro del módulo, y no en el núcleo, o como módulo aparte.

```
# grep -i ext3 .config  
CONFIG_EXT3_FS=m  
CONFIG_EXT3_FS_XATTR=y  
CONFIG_EXT3_FS_POSIX_ACL=y  
CONFIG_EXT3_FS_NFS4ACL=y  
CONFIG_EXT3_FS_SECURITY=y
```

Algunas opciones dependen de otras. A no ser que sepa con exactitud lo que hace, no debería editar el fichero `.config` manualmente para efectuar modificaciones en él, pues corre el riesgo de romper dependencias durante la compilación. Lo más sencillo consiste en pasar por las etapas de configuración expuestas a continuación.

---

 Evite modificar `.config` manualmente. Utilice mejor las herramientas facilitadas desde las fuentes o por su distribución.

---

## b. Recuperar la configuración del núcleo

Puede acceder a la configuración actual del núcleo desde varios sitios. Si un núcleo (o sus fuentes) proceden de un paquete de la distribución, es probable que el fichero `.config` esté ya presente dentro de `/usr/src/linux` o en otro sitio. En este caso, debe consultar la documentación oficial.

En openSUSE, SLES, Red Hat, Debian o Ubuntu por ejemplo, el directorio `/boot` contiene una copia del `.config` que ha servido para la compilación del núcleo.

```
# ls /boot/config*  
/boot/config-2.6.32-5- amd64
```

En este caso, puede reutilizar esta configuración manualmente:

```
# cp /boot/config-2.6.32-5- amd64 /usr/src/linux/.config
```

A menudo se configuran los núcleos con dos opciones interesantes, desgraciadamente ausentes en Debian y Ubuntu.

```
# grep -i KCONF .config  
CONFIG_IKCONFIG=y  
CONFIG_IKCONFIG_PROC=y
```

La primera permite colocar el contenido del `.config` en el propio núcleo durante la compilación. La segunda permite acceder a esta configuración desde el sistema de ficheros virtual `/proc` vía el fichero `/proc/config.gz`. Se comprime el seudofichero en formato gzip. Para leerlo, utilice el comando `zcat`.

```
# zcat /proc/config.gz > /usr/src/linux/.config
```

### c. make oldconfig

El método precedente presenta unos inconvenientes en particular cuando se trata de recuperar la configuración de un núcleo más reciente o más antiguo:

- puede que algunas funcionalidades hayan desaparecido;
- y otras hayan sido añadidas.

En este caso, partir de un fichero de configuración inadecuado puede tener consecuencias nefastas. Para evitar cualquier problema, lo mejor es utilizar una posibilidad que ofrecen las fuentes del núcleo (o más bien Makefile): recuperar la antigua configuración, analizarla, indicar los cambios ocurridos y preguntar qué hacer. Esto se hace vía el comando **make oldconfig**.

En el ejemplo siguiente, se vuelve a coger la configuración de un núcleo 2.6.38 de origen Ubuntu para un núcleo vanilla 3.0-rc7. Hemos truncado muchos de los avisos que aparecen, ya que el núcleo vanilla no contiene ciertas opciones procedentes de parches específicas del núcleo de la distribución. Luego (líneas en negrita), el nuevo núcleo dispone de nuevas posibilidades que no estaban en origen. Debe contestar a cada una de las preguntas, lo que resulta fastidioso.

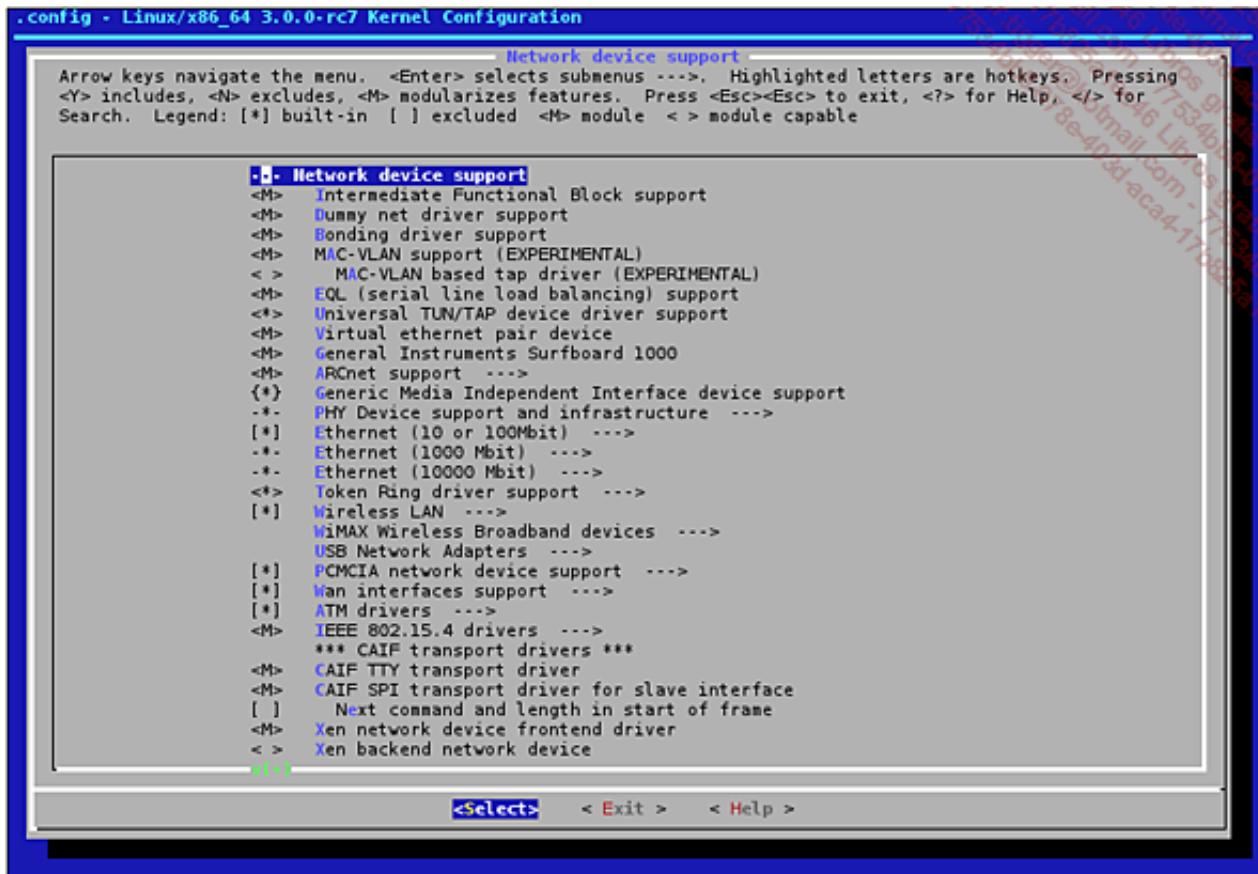
```
x# make oldconfig
scripts/kconfig/conf --oldconfig Kconfig
#
# using defaults found in /boot/config-2.6.38-8-generic
#
/boot/config-2.6.38-8-generic:1185:warning: symbol value 'm' invalid for
BT_L2CAP
/boot/config-2.6.38-8-generic:1186:warning: symbol value 'm' invalid for
BT_SCO
/boot/config-2.6.38-8-generic:3812:warning: symbol value 'm' invalid for
FB_VESA
*
* Restart config...
*
*
* General setup
*
Prompt for development and/or incomplete code/drivers (EXPERIMENTAL)
[Y/n/?] y
Cross-compiler tool prefix (CROSS_COMPILE) []
Local version - append to kernel release (LOCALVERSION) []
Automatically append version information to the version string
(LOCALVERSION_AUTO) [N/y/?] n
Kernel compression mode
> 1. Gzip (KERNEL_GZIP)
   2. Bzip2 (KERNEL_BZIP2)
   3. LZMA (KERNEL_LZMA)
   4. XZ (KERNEL_XZ)
   5. LZO (KERNEL_LZO)
choice[1-5?]: 1
Default hostname (DEFAULT_HOSTNAME) [(none)] (NEW)
```

 Si pulsa la tecla [Intro], se activa la opción por defecto. Puede seguir pulsando la tecla [Intro] hasta el final y luego modificar la configuración con la interfaz textual o gráfica.

### d. make menuconfig

El método anterior es práctico en el caso de una migración. Sin embargo, quizá querrá utilizar algo más ameno. Si no dispone de una interfaz gráfica (a menudo ausente en los servidores, ya que resulta inútil), puede configurar su núcleo pasando por una interfaz en modo consola. Esto quizás recuerde MS-DOS a algunos.

```
# make menuconfig
```



*Las opciones de compilación del núcleo vía menuconfig*

Se incluye el modo de empleo en la interfaz (teclee [Intro], [Y], [N], [M], [?], [/], [Esc]). Observe que si pulsa la tecla [Espacio] se desplegarán todas las opciones posibles. No dude en utilizar la tecla de ayuda para (intentar) entender para qué sirve una opción.

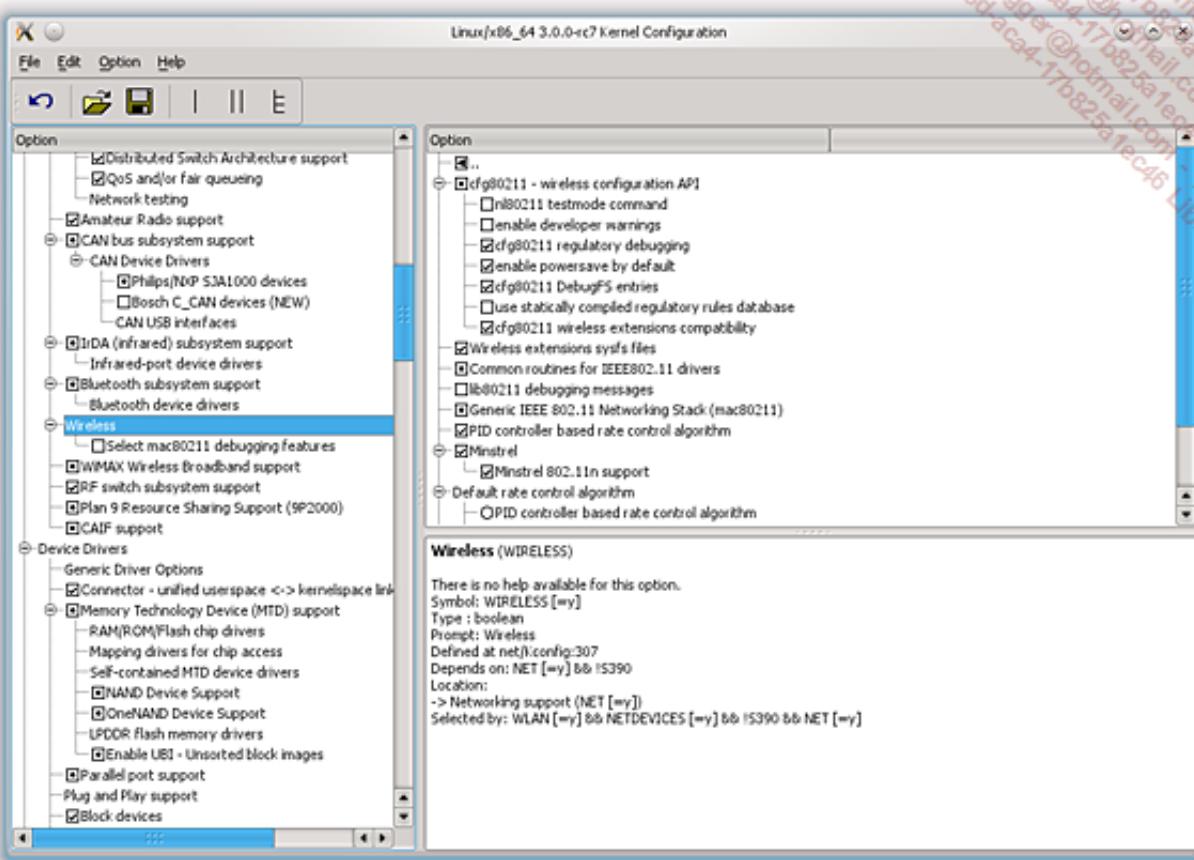
Una vez haya elegido sus opciones, salga y se generará el fichero .config de manera automática.

## e. make xconfig

Si dispone de un entorno gráfico, he aquí la mejor elección posible. Dispone de una interfaz gráfica para configurar su núcleo. La lista de entradas principales está a la izquierda, en forma de árbol. Elige a la derecha las opciones asociadas. En el cuadro inferior, obtiene ayuda para cada una de las opciones. Los menús le permiten efectuar una búsqueda entre los valores.

Desde la interfaz gráfica, puede cargar una configuración procedente de otro fichero de configuración o exportarla hacia un fichero que elija.

```
# make xconfig
```



### *Las opciones de compilación vía xconfig*

Si ejecuta make config, la interfaz utilizada será gtk en vez de qt. Será necesario instalar las librerías de desarrollo gtk.

## f. Algunas opciones de optimización

Las distribuciones Linux suministran núcleos genéricos. Algunas proponen núcleos específicos para algunos tipos de procesadores (Intel, AMD, etc.). En todos los casos, o casi, se facilitan versiones de 32 bits y 64 bits. Estas últimas permiten aprovechar los procesadores Intel o AMD recientes (Athlon64, Prescott, Core2, etc.).

Se conciben estos núcleos para funcionar en un máximo de máquinas, procesadores y dispositivos. Se compilan con opciones estándares y la casi totalidad del soporte de los diferentes periféricos. Así pues, no están optimizados para algunos usos. Le parecerá muy interesante lo que puede aportar la posibilidad de adaptar el núcleo a su máquina:

- Una adecuación perfecta de las optimizaciones del núcleo para su hardware.
- Mayor eficacia según el tipo de uso (servidor, estación de trabajo).
- El soporte de nuevos materiales (núcleo más reciente).
- Una reducción del volumen en disco ocupado por los módulos inútiles.

► En todos los casos, si compila un nuevo núcleo, no desinstale ni borre el antiguo. Consérvelo: en caso de problemas con su núcleo personalizado, tendrá la posibilidad de volver atrás.

Algunas pistas para mejorar su núcleo:

- Active las optimizaciones del núcleo para su procesador. En la entrada **Processor type and features** seleccione, por ejemplo, **Core2 / Newer Xeon** si su máquina dispone de un procesador Core 2 Duo o superior.
- Modifique el valor **Preemption Model** a **Preemptible kernel** y marque la casilla **Preempt the Big Kernel Clock** con el fin de reducir el efecto de latencia del núcleo (para establecer dicho modo en el proceso).
- Modifique el valor del **Timer Frequency** a 300 Hz o 1000 Hz con el fin de reducir el tiempo de respuesta a los eventos.
- En **CPU Frequency Scaling**, también puede aumentar ligeramente la velocidad de inicio del núcleo pasando el valor **Default CPUfreq Governor** a **performance**. Esto no cambia nada de momento. Puede seguir ajustando después la velocidad del procesador.
- Suprime luego los diferentes módulos que no va a necesitar nunca. Por ejemplo, no hace falta conservar las tarjetas Token Ring o ISDN si no las utiliza jamás.

En cuanto a este último punto, tenga cuidado de no suprimir cualquier cosa. No olvide que, si cambia de máquina, a diferencia de Windows, no tiene que volver a instalar obligatoriamente Linux si el núcleo es bastante genérico (quizá tendrá que bootear en single para modificar los módulos cargados en el inicio). El autor migró de un Pentium 4 de hace cuatro años a un Core 2 Duo (con cambio de placa base, tarjeta gráfica, etc.) sin necesidad de reinstalar completamente su sistema. El nuevo sistema puede necesitar módulos desafortunadamente suprimidos.

Piense también en los soportes y periféricos en hotplug o en hotswap. Si por error ha suprimido el driver o una opción del driver, tendrá que recomilar el núcleo o reiniciar a partir del antiguo que haya conservado.

---

 Una buena idea con los núcleos por defecto de las distribuciones consiste en tocar únicamente las opciones de optimización, como las que hemos visto más arriba, dejando todas las otras opciones por defecto.

---

### **¿32 o 64 bits?**

Casi todas las máquinas vendidas hoy en día, incluso los ordenadores portátiles, disponen de un conjunto de instrucciones de 64 bits. Hasta la llegada de Windows 7, los ordenadores disponibles al gran público se vendían con sistemas de 32 bits (Windows XP o Vista, principalmente), a pesar de que existían las versiones de 64 bits. Los 64 bits ofrecen actualmente más ventajas que inconvenientes:

- Una ganancia importante en la gestión de la memoria (adiós a la limitación de 4 GB).
- Una ganancia importante de eficacia en las aplicaciones optimizadas.
- Una garantía de compatibilidad en el futuro.
- Adiós al efecto 2000 (consecuencia de pasar a cero todas las fechas y regresar al 13 de diciembre de 1901) ¡Unix en enero de 2038!

A modo de comparación, la transcodificación de un DVD en DVIX vía dvdrrip (opciones: de una pasada, desenlace inteligente, gran tamaño de redimensionamiento HQ, mp3 192 kbits) se ejecuta a una velocidad de 27 a 30 imágenes por segundo en openSUSE 10.3 en 32 bits (Core 2 Duo e6750). En la misma máquina, la velocidad es de 40-42 imágenes por segundo (en una instalación optimizada para trabajar en 64 bits). O sea, una ganancia media del 25 al 30%. Se constata lo mismo con un cálculo súper-pi optimizado. La ganancia puede ser real, incluso si no es significativa en un entorno ofimático.

Dado que se puede obtener el núcleo de Linux mediante ficheros fuente y que la mayoría de los drivers están incluidos en las distribuciones o en forma de fuentes, cualquier hardware soportado en 32 bits por Linux lo estará también en 64 bits. Se facilitan algunos módulos propietarios (ati, nvidia) en 64 bits. Las distribuciones de 64 bits proporcionan lo necesario (las librerías) para hacer funcionar las aplicaciones de

32 bits de manera nativa.

Si tiene esta posibilidad, pruebe una versión de 64 bits de Linux.

- Si vuelve a compilar un núcleo con las extensiones de 64 bits desde una distribución de 32 bits, únicamente el núcleo en 64 bits funcionará. Habría que instalar luego todas las librerías y todas las herramientas en 64 bits, lo que resulta complejo y largo. En este caso, lo mejor es volver a instalar el sistema.

## 4. Compilación

Ahora que se ha configurado el núcleo, puede iniciar la compilación. Esta etapa es la más larga. El núcleo contiene millones de líneas en lenguaje C y ensamblador. Según las opciones seleccionadas (y, sobre todo, los módulos incluidos), el nivel de ocupación de su máquina y el rendimiento de su procesador, esta etapa requiere desde unos minutos hasta unas horas! En cualquier caso, tiene tiempo para hacer otra cosa.

```
g Kconfig
CHK    include/linux/version.h
UPD    include/linux/version.h
CHK    include/generated/utsrelease.h
UPD    include/generated/utsrelease.h
CC     kernel/bounds.s
GEN    include/generated/bounds.h
CC     arch/x86/kernel/asm-offsets.s
GEN    include/generated/asm-offsets.h
CALL   scripts/checksyscalls.sh
HOSTCC scripts/genksyms/genksyms.o
SHIPPED scripts/genksyms/lex.c
SHIPPED scripts/genksyms/parse.h
SHIPPED scripts/genksyms/keywords.c
HOSTCC scripts/genksyms/lex.o
SHIPPED scripts/genksyms/parse.c
HOSTCC scripts/genksyms/parse.o
HOSTLD scripts/genksyms/genksyms
CC     scripts/mod/empty.o
HOSTCC scripts/mod/mk_elfconfig
MKELF  scripts/mod/elfconfig.h
HOSTCC scripts/mod/file2alias.o
HOSTCC scripts/mod/modpost.o
HOSTCC scripts/mod/sumversion.o
HOSTLD scripts/mod/modpost
HOSTCC scripts/selinux/genheaders/genheaders
HOSTCC scripts/selinux/mdp/mdp
HOSTCC scripts/kallsyms
HOSTCC scripts/commakehash
HOSTCC scripts/recordmcount
CC     init/main.o
CHK    include/generated/compile.h
UPD    include/generated/compile.h
CC     init/version.o
CC     init/do_mounts.o
CC     init/do_mounts_rd.o
CC     init/do_mounts_initrd.o
CC     init/do_mounts_md.o
LD     init-mounts.o
CC     init/initramfs.o
CC     init/calibrate.o
LD     init-built-in.o
HOSTCC usr/gen_init_cpio
GEN    usr/initramfs_data.cpio
```

```

AS      usr/initramfs_data.o
LD      usr/built-in.o
CC      arch/x86/crypto/crc32c-intel.o
LD      arch/x86/crypto/built-in.o
AS [M]  arch/x86/crypto/aes-x86_64-asm_64.o
CC [M]  arch/x86/crypto/aes_glue.o
AS [M]  arch/x86/crypto/aesni-intel_asm.o
CC [M]  arch/x86/crypto/aesni-intel_glue.o
CC [M]  arch/x86/crypto/fpu.o
AS [M]  arch/x86/crypto/ghash-clmulni-intel_asm.o
CC [M]  arch/x86/crypto/ghash-clmulni-intel_glue.o
AS [M]  arch/x86/crypto/salsa20-x86_64-asm_64.o
CC [M]  arch/x86/crypto/salsa20_glue.o
AS [M]  arch/x86/crypto/twofish-x86_64-asm_64.o
CC [M]  arch/x86/crypto/twofish_glue.o
LD [M]  arch/x86/crypto/aes-x86_64.o
LD [M]  arch/x86/crypto/twofish-x86_64.o
LD [M]  arch/x86/crypto/salsa20-x86_64.o
...
BUILD  arch/x86/boot/bzImage
Root device is (8, 6)
Setup is 11012 bytes (padded to 11264 bytes).
System is 1567 kB
Kernel: arch/x86/boot/bzImage is ready (#1)
Building modules, stage 2.
MODPOST 1950 modules
...
LD [M]  sound/usb/snd-usb-lib.ko
CC      sound/usb/usx2y/snd-usb-usx2y.mod.o
LD [M]  sound/usb/usx2y/snd-usb-usx2y.ko

```

La primera columna representa la acción que se realiza en el fichero representado por la segunda columna. CC indica una compilación; LD, la edición de los vínculos; una [M], la acción sobre un módulo, etc.

## 5. Instalación

Si se efectúa la compilación sin error, le quedan dos acciones por hacer. Primero instale los módulos. El comando siguiente los instala en `/lib/module/<version_núcleo>` y crea el fichero de las dependencias asociadas.

```

# make modules_install|more
INSTALL arch/x86/crypto/aes-x86_64.ko
INSTALL arch/x86/crypto/aesni-intel.ko
INSTALL arch/x86/crypto/ghash-clmulni-intel.ko
INSTALL arch/x86/crypto/salsa20-x86_64.ko
INSTALL arch/x86/crypto/twofish-x86_64.ko
INSTALL arch/x86/kernel/cpu/mcheck/mce-inject.ko
INSTALL arch/x86/kernel/cpuid.ko
INSTALL arch/x86/kernel/microcode.ko
INSTALL arch/x86/kernel/msr.ko
INSTALL arch/x86/kvm/kvm-amd.ko
INSTALL arch/x86/kvm/kvm-intel.ko
INSTALL arch/x86/kvm/kvm.ko
INSTALL arch/x86/oprofile/oprofile.ko
...
INSTALL /lib/firmware/yam/1200.bin
INSTALL /lib/firmware/yam/9600.bin
DEPMOD 3.0.0-rc7

```

Este segundo comando vuelve a copiar el núcleo y lo necesario asociado en `/boot`. Según las distribuciones, va a crear también el initrd asociado y a modificar la configuración del cargador de inicio GRUB.

```

# make install
sh /usr/src/linux-3.0-rc7/arch/x86/boot/install.sh 3.0.0-rc7
arch/x86/boot/bzImage \
    System.map "/boot"
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 3.0.0-rc7
/boot/vmlinuz-3.0.0-rc7
run-parts: executing /etc/kernel/postinst.d/nvidia-common 3.0.0-rc7
/boot/vmlinuz-3.0.0-rc7
run-parts: executing /etc/kernel/postinst.d/pm-utils 3.0.0-rc7
/boot/vmlinuz-3.0.0-rc7
run-parts: executing /etc/kernel/postinst.d/update-notifier 3.0.0-rc7
/boot/vmlinuz-3.0.0-rc7
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 3.0.0-rc7
/boot/vmlinuz-3.0.0-rc7
Generating grub.cfg ...
Found linux image: /boot/vmlinuz-3.0.0-rc7
Found linux image: /boot/vmlinuz-3.0.0-rc7.old
Found linux image: /boot/vmlinuz-2.6.38-10-generic
Found initrd image: /boot/initrd.img-2.6.38-10-generic
Found memtest86+ image: /memtest86+.bin
Found Windows 7 (loader) on /dev/sdal
done

```

Observe que no se ha reconstruido el initrd en este caso, lo que no es normal y debería funcionar con la salida de un núcleo estable. Puede reconstruirlo usted mismo, por ejemplo en Debian o Ubuntu:

```

# cd /boot
# mkinitramfs -o initrd.img-3.0.0-rc7 3.0.0-rc7
# update-grub2

```

Respete el orden precedente. Si instala el núcleo antes que sus módulos, no se podrá volver a construir el initrd, ya que aún no se han instalado los módulos que deberían estar presentes. Puede que no se genere el initrd por defecto. En este caso, tendrá que volver a ejecutar este comando después de la instalación con los parámetros correspondientes al núcleo instalado.

```

# ls -l /boot/*3.0.0*
-rw-r--r-- 1 root root 133632 2011-07-20 22:28 /boot/config-3.0.0-rc7
-rw-r--r-- 1 root root 148724569 2011-07-20 22:29 /boot/initrd.img-3.0.0-rc7
-rw-r--r-- 1 root root 2560178 2011-07-20 22:28 /boot/System.map-3.0.0-rc7
-rw-r--r-- 1 root root 4564832 2011-07-20 22:28 /boot/vmlinuz-3.0.0-rc7

```

Verifique si se ha modificado la configuración del cargador de lanzamiento con GRUB (y no GRUB2).

```

# grep 2.6.24 /boot/grub/menu.lst
title openSUSE 11.4 - 3.0.0-rc7
kernel /boot/vmlinuz-3.0.0-rc7 root=/dev/disk/by-id/scsi-
SATA_ST380011A_5JVTH798-part6 vga=0x31a resume=/dev/sda5 splash=silent
showopts
...

```



No tiene por qué reinstalar GRUB o GRUB2 cuando modifica su fichero de configuración.

Si utiliza LILO, además de modificar `/etc/lilo.conf`, debe reinstalarlo con `/sbin/lilo`.

## 6. Test

Si todas las etapas precedentes han culminado con éxito, sólo le queda encender de nuevo su ordenador y seleccionar su nuevo núcleo en el momento de la carga. Si el boot termina correctamente (acceso a la consola o entorno gráfico), abra una consola y verifique la versión de su sistema.

```
$ uname -a
Linux slyserver 3.0.0-rc7 #1 SMP Wed Jul 20 19:20:16
CEST 2011 x86_64 x86_64 x86_64 GNU/Linux
```

## 7. Otras opciones

Después de la compilación los ficheros intermediarios (ficheros objetos) toman mucho espacio. Para suprimirlos, utilice el comando siguiente: `# make clean`

Si su distribución cuenta con el sistema rpm de paquetería, puede crear los paquetes del núcleo (fuentes, headers, núcleo) con el comando siguiente: `# make rpm`

# Los ficheros periféricos

## 1. Introducción

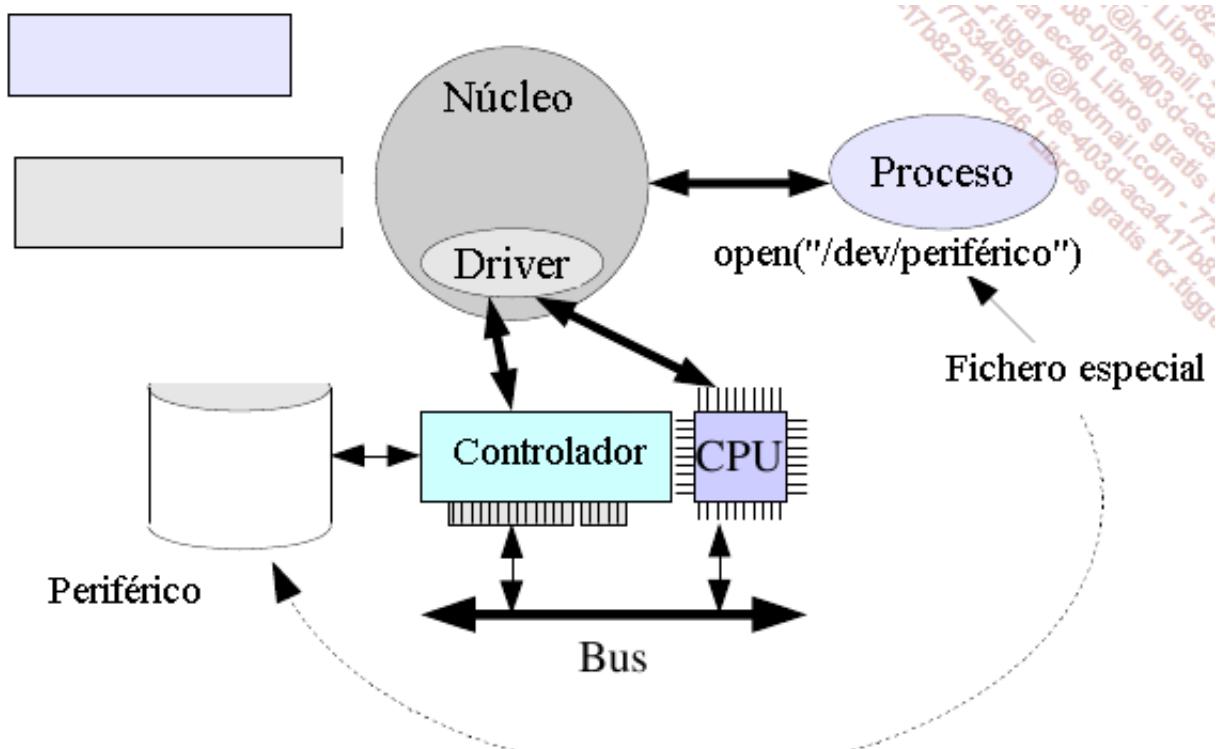
Regresamos al funcionamiento de los periféricos dentro de un ordenador. El principio suele ser el mismo en todos los ordenadores.

Los periféricos están vinculados a un controlador, por ejemplo IDE o SATA para los discos IDE, un controlador SCSI para los discos, lectores y otros escáneres SCSI, o incluso un controlador USB. Un controlador debe saber controlar varios periféricos vinculados a él.

El controlador se comunica con el microprocesador y la memoria con la ayuda del bus (bus de comandos y datos).

En cuanto a Linux, se gestiona el controlador y sus periféricos con la ayuda de drivers (un driver para el controlador, y uno o varios drivers para los periféricos relacionados con él. Por ejemplo un driver para el controlador SCSI, luego un driver para los discos, y otro para los escáneres, y otro para un CD-Rom). El driver suele ser un módulo complementario del núcleo, entregado por el fabricante o ya presente.

Los periféricos son ficheros. En consecuencia, los procesos acceden a los periféricos mediante estos ficheros con la ayuda de las funciones en lenguaje C cuyo código está en el núcleo. El proceso debe abrir primero el fichero especial del periférico (primitive open), luego leer (read) o escribir (write) datos desde o hacia el periférico como lo haría un fichero normal. Luego el driver del periférico interpreta estas operaciones de lectura/escritura.



*Linux accede a los periféricos mediante unos ficheros especiales*

En el fichero especial `/dev/periférico`, el sistema de gestión de ficheros encuentra la información necesaria para dirigirse al driver correspondiente a través del periférico abierto por un proceso.

## 2. Ficheros especiales

Por convención, se colocan los ficheros especiales periféricos en el directorio /dev; éstos disponen de un inodo único, como cualquier otro fichero. Por lo tanto, puede conocer sus atributos mediante el comando **ls -l**.

El primer carácter identifica el tipo de periférico:

- **c**: tipo de periférico en modo carácter;
- **b**: tipo de periférico en modo bloque.

Estos modos diferencian el tipo de intercambio de datos entre el módulo de gestión de ficheros y el driver del periférico. En modo carácter, no se utilizan los buffers del sistema y el intercambio se hace byte por byte.

En modo bloque, el sistema accede al periférico mediante un índice que representa las coordenadas del bloque de datos en el soporte. Por lo tanto, es más rápido para periféricos del tipo discos duros.

Los otros dos atributos esenciales de un fichero periférico son el par de información que se encuentra en lugar del tamaño del fichero: el número **mayor** y el número **menor**.

- El número mayor identifica el driver y como consecuencia el controlador de periférico.
- El número menor suele identificar el periférico, pero también puede designar una particularidad del periférico, como la partición de un disco, una ubicación concreta, el número de tarjeta (en caso de que haya varias tarjetas controladores idénticas, varias tarjetas sonido, etc.).

Veamos algunos ficheros especiales corrientes, según la distribución:

- **/dev/mem**: la memoria física.
- **/dev/kmem**: la memoria virtual del núcleo.
- **/dev/console**: la consola maestra (/dev/syscon).
- **/dev/tty**: la entrada/salida estándar del proceso en curso.
- **/dev/mouse**: el ratón, a menudo un atajo.
- **/dev/swap**: el disco swap primario.
- **/dev/null**: la basura UNIX. Se puede escribir en ella. La lectura provoca un EOF.
- **/dev/root**: sistema de fichero especial root.
- **/dev/dump**: el disco donde el núcleo hace su dump en caso de «panic system».
- **/dev/rmt0**: lector de banda magnética o de cartucho en modo carácter.
- **/dev/fd0**: lector de disquetes en modo bloque.
- **/dev/pts/1**: ídem, pero para Unix SYSTEM V (y Linux).
- **/dev/lp0**: una impresora paralela.
- **/dev/ttys0**: puerto COM1.
- **/dev/ttys1**: puerto COM2.
- **/dev/psaux**: puerto PS2 para el ratón.
- **/dev/sound**: tarjeta de sonido.
- **/dev/dsp**: controlador DSP de la tarjeta de sonido.

- `/dev/sequencer`: secuenciador MIDI de la tarjeta de sonido.
- `/dev/ide/*`: los periféricos IDE.
- `/dev/scsi/*`: los periféricos SCSI.
- `/dev/usb/*`: los periféricos USB.
- `/dev/hdX`: los discos IDE.
- `/dev/sdX`: los discos SATA o SCSI.
- etc.

### 3. Crear un fichero especial

El comando **mknod** permite crear un fichero especial. Aunque Linux disponga de métodos particulares que a menudo se encargan automáticamente de la creación de los ficheros periféricos (`udev`), en ocasiones la documentación del periférico especifica cómo configurarlo manualmente. En este caso, ¡cuidado! El sistema de ficheros `/dev` suele ser de tipo **udev** y totalmente dinámico! Habrá que encontrar otra solución.

```
mknod /dev/periférico tipo mayor menor
```

O bien se integran los drivers en el núcleo durante la compilación de éste, o se compilan estos drivers en forma de módulos complementarios cargados de manera dinámica. Según las distribuciones, el directorio `/dev` resulta ser a veces un sistema de ficheros dinámico (`devfs`, `udev`) cuyo contenido cambia en función de la presencia o no de los periféricos. Así, el driver del periférico y un demonio particular **devfsd** o **udevdse** encargan de la creación del fichero periférico. Esto facilita, por ejemplo, la conexión en caliente, como la conexión de periféricos USB bajo demanda: el núcleo detecta el dispositivo, carga el driver correcto y este driver crea dinámicamente el fichero periférico.

A veces, sólo una parte del directorio `/dev` es dinámica, como el soporte del USB con el sistema de ficheros **usbdevfs**.

## 4. Conocer su hardware

### a. Bus PCI

El comando **lspci** proporciona la información detallada sobre las tarjetas y los adaptadores relacionados con el bus PCI. Los adaptadores pueden ser los conectados a los puertos de extensión de la placa base pero también los integrados a la placa base (controladores IDE/SATA, tarjetas de red, etc.). Los AGP son considerados bus PCI.

```
# lspci
00:00.0 Host bridge: ATI Technologies Inc RS480 Host Bridge (rev 10)
00:01.0 PCI bridge: ATI Technologies Inc RS480 PCI Bridge
00:11.0 IDE interface: ATI Technologies Inc 437A Serial ATA Controller
00:12.0 IDE interface: ATI Technologies Inc 4379 Serial ATA Controller
00:13.0 USB Controller: ATI Technologies Inc IXP SB400 USB Host Controller
00:13.1 USB Controller: ATI Technologies Inc IXP SB400 USB Host Controller
00:130,2 USB Controller: ATI Technologies Inc IXP SB400 USB2 Host Controller
00:14.0 SMBus: ATI Technologies Inc IXP SB400 SMBus Controller (rev 10)
00:14.1 IDE interface: ATI Technologies Inc Standard Dual Channel PCI IDE
Controller
00:14.3 ISA bridge: ATI Technologies Inc IXP SB400 PCI-ISA Bridge
00:14.4 PCI bridge: ATI Technologies Inc IXP SB400 PCI-PCI Bridge
```

```
00:14.5 Multimedia audio controller: ATI Technologies Inc IXP SB400
AC'97 Audio Controller (rev 01)
00:18.0 Host bridge: Advanced Micro Devices [AMD] K8 [Athlon64/Opteron]
HyperTransport Technology Configuration
00:18.1 Host bridge: Advanced Micro Devices [AMD] K8 [Athlon64/Opteron]
Address Map
00:18.2 Host bridge: Advanced Micro Devices [AMD] K8 [Athlon64/Opteron]
DRAM Controller
00:18.3 Host bridge: Advanced Micro Devices [AMD] K8 [Athlon64/Opteron]
Miscellaneous Control
01:05.0 VGA compatible controller: ATI Technologies Inc RS480 [Radeon
Xpress 200G Series]
02:03.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8169
Gigabit Ethernet (rev 10)
```

Puede ir aún más lejos en los detalles con la opción **-v** y especificar un adaptador con sus identificadores. Para obtener la información detallada sobre el controlador Ethernet (02:03.0), haga como sigue:

```
# lspci -v -s 02:03.0
02:03.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8169
Gigabit Ethernet (rev 10)
    Subsystem: Unknown device 1631:d008
    Flags: bus master, 66MHz, medium devsel, latency 64, IRQ 20
    I/O ports at 8800 [size=256]
    Memory at ff3ffc00 (32-bit, non-prefetchable) [size=256]
    Expansion ROM at ff300000 [disabled] [size=128K]
    Capabilities: [dc] Power Management version 2
```

Cuando es posible, la línea Subsystem indica el fabricante y el modelo exacto del periférico. Se identifica el periférico con un par de valores hexadecimales. El primero es el identificador único del fabricante (vendor). El segundo es el identificador del modelo. Algunos modelos tienen a veces los mismos identificadores, aunque el dispositivo difiera (cambio de chipset). La correspondencia es estática (se inscriben las asociaciones dentro del código compilado). Sirve, entre otras cosas, para la carga de los módulos correctos. Si la información no es suficiente, especifique la opción **-vv**.

## b. Bus USB

El comando **lsusb** hace lo mismo que lspci, pero para el bus USB:

```
# lsusb
Bus 003 Device 002: ID 04b4:6830 Cypress Semiconductor Corp. USB-2.0 IDE
Adapter
Bus 003 Device 001: ID 0000:0000
Bus 008 Device 002: ID 126f:0161 TwinMOS
Bus 008 Device 001: ID 0000:0000
Bus 006 Device 001: ID 0000:0000
Bus 007 Device 001: ID 0000:0000
Bus 005 Device 001: ID 0000:0000
Bus 001 Device 001: ID 0000:0000
Bus 004 Device 003: ID 046d:092e Logitech, Inc.
Bus 004 Device 002: ID 046d:c50e Logitech, Inc. MX-1000 Cordless Mouse
Receiver
Bus 004 Device 001: ID 0000:0000
Bus 002 Device 002: ID 045e:00dd Microsoft Corp.
Bus 002 Device 001: ID 0000:0000
```

Cuando es posible, Linux indica cuáles son los nombres de los periféricos mediante una base de identificadores, de la misma manera que para las tarjetas PCI. Estos identificadores también sirven para determinar qué driver USB cargar.

Como en el caso de **lspci**, obtendrá más información con **-v** y con **-d**:

```
# lsusb -d 046d:092e
Bus 004 Device 003: ID 046d:092e Logitech, Inc.
# lsusb -v -d 045e:00dd

Bus 002 Device 002: ID 045e:00dd Microsoft Corp.
Device Descriptor:
  bLength          18
  bDescriptorType   1
  bcdUSB         2.00
  bDeviceClass      0 (Defined at Interface level)
  bDeviceSubClass    0
  bDeviceProtocol    0
  bMaxPacketSize0     8
  idVendor        0x045e Microsoft Corp.
  idProduct        0x00dd
  bcdDevice        1.73
  iManufacturer      1 Microsoft
  iProduct          2 Comfort Curve Keyboard 2000
  iSerial           0
  bNumConfigurations 1
...
...
```

### c. Recursos físicos

El sistema de ficheros virtual /proc está lleno de información sobre su hardware. Veamos una lista no exhaustiva.

**Interrupciones** **Canales DMA** **Intervalos de direcciones de entradas-salidas**

```
# cat /proc/interrupts
          CPU0      CPU1
 0:    692294      559242  IO-APICT-edge      timer
 1:        1          1  IO-APICT-edge      i8042
 8:        1          0  IO-APICT-edge      rtc
 9:        0          0  IO-APICT-fasteoi    acpi
 12:        0          4  IO-APICT-edge      i8042
 16:      591      430773  IO-APICT-fasteoi  ahci, uhci_hcd:usb1,
ohci1394, nvidia
 17:      39      15025  IO-APICT-fasteoi  libata
 18:      285      39829  IO-APICT-fasteoi  ehci_hcd:usb3,
uhci_hcd:usb4, uhci_hcd:usb7
 19:        0          0  IO-APICT-fasteoi  uhci_hcd:usb6
 21:     11107          28  IO-APICT-fasteoi  uhci_hcd:usb2
 22:     397327      3429  IO-APICT-fasteoi  libata, libata, HDA
Intel
 23:        1      2533  IO-APICT-fasteoi  uhci_hcd:usb5,
ehci_hcd:usb8
4347:        40      24310 PCI-MSI-edge      eth0
 NMI:        0          0
 LOC:     1251342     1251335
ERR:        0
```

```
# cat /proc/dma
4: cascada
```

```
# cat /proc/ioports
0000-001f: dma1
0020-0021: pic1
```

```

0040-0043: timer0
0050-0053: timer1
0060-006f: keyboard
0070-0077: rtc
0080-008f: dma_page reg
00a0-00a1: pic2
00c0-00df: dma2
00f0-00ff: fpu
0290-0297: pnp 00:06
    0295-0296: w83627ehf
03c0-03df: vesafb
03f8-03ff: serial
0400-041f: 0000:00:1f.3
    0400-041f: i801_smbus
0480-04bf: 0000:00:1f.0
0800-087f: 0000:00:1f.0
    0800-0803: ACPI PM1a_EVT_BLK
    0804-0805: ACPI PM1a_CNT_BLK
    0808-080b: ACPI PM_TMR
    0810-0815: ACPI CPU throttle
    0820-082f: ACPI GPE0_BLK
0cf8-0cff: PCI conf1
9400-940f: 0000:00:1f.2
9480-948f: 0000:00:1f.2
    9480-948f: libata
...

```

### **Periféricos (bloque, carácter) Particiones Procesadores**

```

# cat /proc/devices
Character devices:
  1 mem
  2 pty
  3 ttyp
  4 /dev/vc/0
  4 tty
  4 ttyS
...
  14 sound
...
 171 ieee1394
180 usb
189 usb_device
195 nvidia
253 rtc
254 usb_endpoint

Block devices:
  7 loop
  8 sd
  9 md
11 sr
65 sd
...
253 device-mapper
254 mdp

```

```

# cat /proc/partitions
major minor #blocks name

  8      0   160836480 sda
  8      1   41945683 sda1
  8      2     530145 sda2

```

```

8      3    41945715 sda3
8      4          1 sda4
8      5    2104483 sda5
8      6    74308626 sda6
8     16   156290904 sdb
8     17   156288321 sdb1
8     32   293036184 sdc
8     33   293033601 sdc1
8     48   503808 sdd
8     49   503792 sdd1

```

```

# cat /proc/cpuinfo
processor      : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 15
model name     : Intel(R) Core(TM)2 Duo CPU      E6750 @ 2.66GHz
stepping        : 11
cpu MHz         : 1998,000
cache size      : 4.096 KB
physical id     : 0
siblings         : 2
core id          : 0
cpu cores        : 2
fpu              : yes
fpu_exception    : yes
cpuid level     : 10
wp               : yes
flags            : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr
pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm
syscall nx lm constant_tsc pni monitor ds_cpl vmx smx est tm2 ssse3
cx16 xtpr lahf_lm
bogomips        : 6804.61
clflush size     : 64
cache_alignment  : 64
address sizes    : 36 bits physical, 48 bits virtual
power management:
...

```

## d. Otras herramientas

Las distintas distribuciones entregan juegos de herramientas complementarias, como **hwinfo** o **dmidecode**.

### hwinfo

La herramienta **hwinfo** detecta su hardware y le proporciona una lista de éste (de manera corta con la opción `--short`). Este comando realiza un rastreo por todo su hardware, que devuelve la información:

```

# hwinfo --short
cpu:                      Intel(R) Core(TM)2 Duo CPU      E6750 @ 2.66GHz, 2000 MHz
...
keyboard:                 /dev/input/event4  Microsoft Comfort Curve Keyboard 2000
mouse:                    /dev/input/mice    Logitech MX-1000 Cordless Mouse Receiver
monitor:                  Generic Monitor
graphics card:             ASUSTeK GeForce 8600 GT
sound:                    ASUSTeK 82801I (ICH9 Family) HD Audio Controller

```

```

storage: ASUSTeK 82801IB (ICH9) 2 port SATA IDE Controller
...
network:
  eth0 ASUSTeK L1 Gigabit Ethernet Adapter
...
disk:
  /dev/sda HDT722516DLA380
...
partition:
  /dev/sdal Partition
...
cdrom:
  /dev/sr0 TSSTcorp CD/DVDW SH-S183A
...
usb controller:
  ASUSTeK 82801I (ICH9 Family) USB UHCI Controller #4
...
BIOS:
  BIOS
bridge:
  ASUSTeK 82G33/G31/P35/P31 Express DRAM Controller
...
Port:
  Intel 82801I (ICH9 Family) PCI Express Port 1
...
memory:
  Main Memory
firewire controller:
  ASUSTeK IEEE 1394 Host Controller
unknown:
...
  Keyboard controller
...
  /dev/ttyS0 16550A
  Logitech Camera
  /dev/input/event5 Microsoft Comfort Curve Keyboard 2000

```

Para obtener más detalles, suprima la opción `--short` y especifique si es preciso qué componente quiere detallar. Por ejemplo, `--cpu` para el procesador, `--memory` para la memoria, etc. La lista está en el manual del comando.

```

# hwinfo --cpu
01: None 00.0: 10103 CPU
[Created at cpu.301]
Unique ID: rdCR.j8NaKXDZtZ6
Hardware Class: cpu
Arch: X86-64
Vendor: "GenuineIntel"
Model: 6.15.11 "Intel(R) Core(TM)2 Duo CPU      E6750 @ 2.66GHz"
Features:
fpu,vme,de,pse,tsc/msr,pae,mce,cx8,apic,sep,mtrr,pge,mca,cmov,pat,pse36,
clflush,dts,acpi,mmx,fxsr,sse,sse2,ss,ht,tm,syscall,nx,lm,constant_tsc,
pni,monitor,ds_cpl,vmx,smx,est,tm2,ssse3,cx16,xtpr,lahf_lm
Clock: 2666 MHz
BogoMips: 6804.61
Cache: 4096 kb
Units/Processor: 2
Config Status: cfg=new, avail=yes, need=no, active=unknown
...
# hwinfo --memory
01: None 00.0: 10102 Main Memory
[Created at memory.61]

```

```
Unique ID: rdCR.CxwsZFjVASF
Hardware Class: memory
Model: "Main Memory"
Memory Range: 0x00000000-0x7ff7ffff (rw)
Memory Size: 2 GB
Config Status: cfg=new, avail=yes, need=no, active=unknown
```

## dmidecode

La herramienta **dmidecode** no interroga a los dispositivos directamente, sino que lee e interpreta la tabla **DMI** (*Desktop Management Interface*) del ordenador, a veces llamada también **SMBIOS** (*System Management BIOS*). Proporciona no solamente información sobre el estado físico actual de la máquina, sino también sobre sus extensiones posibles (por ejemplo, la velocidad máxima del procesador, la cantidad de memoria posible, etc.). A diferencia de **hwinfo**, que interroga a un componente, por ejemplo la CPU, **dmidecode** lee la información tal como la detecta la BIOS y la placa base. Es rápido, a veces más concreto que **hwinfo**, pero en ocasiones da errores, por lo que se recomienda comprobarlo.

Como la salida es mucho más larga que con **hwinfo**, especifique qué información desea con **-s** o **-t** (consulte el manual):

```
# dmidecode -t processor
# dmidecode 2.9
SMBIOS 2.4 present.

Handle 0x0004, DMI type 4, 35 bytes
Processor Information
    Socket Designation: LGA775
    Type: Central Processor
    Family: Pentium 4
    Manufacturer: Intel
    ID: FB 06 00 00 FF FB EB BF
    Signature: Type 0, Family 6, Model 15, Stepping 11
    Flags:
        FPU (Floating-point unit on-chip)
        VME (Virtual mode extension)
        DE (Debugging extension)
        PSE (Page size extension)
        TSC (Time stamp counter)
        MSR (Model specific registers)
        PAE (Physical address extension)
        MCE (Machine check exception)
        CX8 (CMPXCHG8 instruction supported)
        APIC (On-chip APIC hardware supported)
        SEP (Fast system call)
        MTRR (Memory type range registers)
        PGE (Page global enable)
        MCA (Machine check architecture)
        CMOV (Conditional move instruction supported)
        PAT (Page attribute table)
        PSE-36 (36-bit page size extension)
        CLFLUSH (CLFLUSH instruction supported)
        DS (Debug store) ACPI (ACPI supported)
        MMX (MMX technology supported)
        FXSR (Fast floating-point save and restore)
        SSE (Streaming SIMD extensions)
        SSE2 (Streaming SIMD extensions 2)
        SS (Self-snoop)
        HTT (Hyper-threading technology)
        TM (Thermal monitor supported)
        PBE (Pending break enabled)
Version: Intel(R) Core(TM)2 Duo CPU E6750 @ 2.66GHz
Voltage: 1.3 V
```

```
External Clock: 427 MHz
Max Speed: 3800 MHz
Current Speed: 3416 MHz
Status: Populated, Enabled
Upgrade: Socket LGA775
L1 Cache Handle: 0x0005
L2 Cache Handle: 0x0006
L3 Cache Handle: 0x0007
Serial Number: To Be Filled By O.E.M.
Asset Tag: To Be Filled By O.E.M.
Part Number: To Be Filled By O.E.M.
```

Observe que los valores en negrita no son los mismos que los devueltos por los otros comandos. Aquí no hay error: el procesador está «overclockeado» a 3,4 GHz...

```
# dmidecode -s processor-frequency
3416 MHz
```

## 5. El soporte del USB y del hotplug

### a. Los módulos

El **USB** (*Universal Serial Bus*) es un bus de datos en modo serie y «plug and play». El principio consiste en que, una vez conectado el adaptador USB, el sistema carga el posible driver correspondiente y el equipo funciona en seguida. En USB 2.0, las tasas de transferencia pueden alcanzar 480 Mbits/s. El futuro USB 3 debería permitir una tasa de velocidad de 4,8 Gbits/s.

Para las versiones USB 1.0 y USB 1.1, dos tipos de controladores se reparten el mercado: **UHCI** y **OHCI**.

- Universal Controller Host Interface: desarrollado por Intel.
- Open Controller Host Interface: los demás.

En USB 2.0, el controlador se llama **EHCI**: *Enhanced Host Controller Interface*.

Linux gestiona los tres tipos de controlador.

El módulo básico se llama **usbcore**. Propone el conjunto de las API necesarias para otros módulos:

- **ohci\_hcd**: soporte OHCI.
- **uhci\_hcd**: soporte UHCI.
- **ehci\_hcd**: soporte EHCI.
- **usb\_storage**: capa de acceso a los soportes masivos: discos externos, claves USB, etc.
- **usbhid**: capa de acceso al soporte de los periféricos **HID** (*Human Interface Device*) del tipo teclados, ratones, joystick, etc.
- **snd-usb-audio**: soporte de tarjetas de sonido USB.
- **usbvideo**: soporte de tarjetas de vídeo y adquisición de USB.
- **irda-usb**: soporte de los puertos de infrarrojo USB.
- **usbnet**: soporte de las tarjetas de red USB.
- etc.

## b. Carga

Se cargan los módulos USB (y otros relacionados con el hardware) de distintas formas:

- Mediante el **ramdisk initial**: actualmente la mayoría de los teclados y ratones son de tipo USB. Por eso, se suele cargar el soporte básico del USB y el módulo usbhid mediante el initrd (Initial Ramdisk). De ahí que a menudo haya un pequeño desfase entre el momento en el cual GRUB carga el núcleo y el momento en el que es posible utilizar el teclado, entre el final del soporte del USB por la BIOS (paso del núcleo al modo protegido) y el momento en que el núcleo se encarga del soporte USB.
- Vía **init**: un servicio encargado de la detección del hardware y de la carga de los drivers carga la lista de los módulos correspondiente al hardware.
- Vía **kmod**: carga automática de los módulos del núcleo. Cuando el núcleo detecta la presencia de un nuevo periférico USB, puede cargar el módulo correspondiente. Recuerde que un módulo proporciona los identificadores del hardware del cual se encarga. El fichero **modules.usbmapprove**ee esta correspondencia. Entonces el núcleo ejecuta **modprobe**.
- Vía **udev o hotplug**: unas reglas permiten especificar acciones cuando llegan nuevos periféricos, entre las cuales por ejemplo, se encuentra la carga de módulos complementarios.
- Manualmente.

## c. hotplug, usbmgr

Antes de la llegada de udev y HAL, se utilizaba principalmente el proyecto **Linux Hotplug Project**. hotplug no gestionaba únicamente el USB, sino cualquier hardware conectado, en caliente o no (coldplug), al PC.

Ya no se utiliza hotplug en ninguna de las distribuciones recientes basadas en el núcleo 2.6. Ha dejado su sitio al tandem udev/HAL.

**Usbmgr** era un sistema de hotplug previsto para la gestión del USB. Ya no se utiliza.

## d. udev

**udev** es un sistema de ficheros dinámico que sustituye al antiguo **devfs** de los núcleos 2.4. **Udev**gestiona todo el árbol /dev. Va a crear y modificar los ficheros periféricos presentes en este directorio.

A diferencia de hotplug, que se ejecutaba en modo kernel, se inicia **udev** como un servicio clásico, asociado a un sistema de ficheros particular. Se sitúa en el espacio del usuario.

```
# mount | grep udev  
udev on /dev type tmpfs (rw,mode=0755)
```

El

núcleo gestiona eventos y mensajes. **udev** lee los mensajes emitidos por el núcleo y los interpreta. Dispone de reglas que aplica según el tipo de mensaje. Por ejemplo, veamos tres reglas sencillas:

```
KERNEL=="raw1394*",           GROUP="video"  
KERNEL=="dv1394*",           SYMLINK+="dv1394/%n", GROUP="video"  
KERNEL=="video1394*",         SYMLINK+="video1394/%n", GROUP="video"
```

**KERNEL**: nombre del evento del núcleo. Aquí aparece un directorio Firewire.

- **GROUP**: el fichero periférico pertenecerá al grupo indicado.
- **SYMLINK**: udev va a crear un vínculo simbólico del fichero periférico hacia la ubicación indicada. Aquí el %n representa el número en el orden de detección.

Es sólo un ejemplo. La formación LPI no requiere escribir reglas elaboradas, sino entender el mecanismo asociado. Con las reglas anteriores, si usted conecta una videocámara digital gracias al enchufe DV a su tarjeta Firewire, el núcleo va a generar un evento cuyo nombre empieza por «vídeo1394». Se ejecutará la regla correspondiente:

- Aparición de /dev/video1394.
- Modificación de su grupo en vídeo.
- Creación de un vínculo simbólico entre /dev/video1394 y /dev/video1394/0 (el primero).

Se colocan las reglas en `/etc/udev/rules.d`.

```
$ ls -l
total 180
-rw-r--r-- 1 root root 191 sep 21 2007 05-udev-early.rules
-rw-r--r-- 1 root root 366 oct  2 2007 40-alsa.rules
-rw-r--r-- 1 root root 2276 sep 24 2007 40-bluetooth.rules
-rw-r--r-- 1 root root 4937 sep 21 2007 50-udev-default.rules
-rw-r--r-- 1 root root 571 sep 22 2007 51-lirc.rules
-rw-r--r-- 1 root root 399 oct 10 2007 55-hpmud.rules
-rw-r--r-- 1 root root 83444 feb 17 14:22 55-libsane.rules
-rw-r--r-- 1 root root 119 sep 21 2007 56-idedma.rules
-rw-r--r-- 1 root root 119 sep 21 2007 60-cdrom_id.rules
-rw-r--r-- 1 root root 1424 sep 21 2007 60-persistent-input.rules
-rw-r--r-- 1 root root 4347 sep 21 2007 60-persistent-storage.rules
-rw-r--r-- 1 root root 918 sep 21 2007 64-device-mapper.rules
-rw-r--r-- 1 root root 725 sep 21 2007 64-md-raid.rules
-rw-r--r-- 1 root root 1290 sep 22 2007 70-kpartx.rules
-rw-r--r-- 1 root root 611 oct  5 2007 70-persistent-cd.rules
-rw-r--r-- 1 root root 325 oct  5 2007 70-persistent-net.rules
...
...
```

Usted puede modificar las reglas y crear las suyas propias. En este caso, cree un fichero llamado **99-local.rules** (por ejemplo) y coloque dentro sus reglas.

A modo de ejemplo, las diferentes rutas posibles de acceso a los discos y particiones, presentadas en el capítulo Los discos y el sistema de ficheros - Acceder a los sistemas de ficheros, proceden de reglas udev. La regla siguiente (que depende de otras reglas en amont, lo que se puede deducir por el test de una variable **udev** DETYPE ya ubicada) importa el resultado de un comando **vol\_id** que usted ya conoce para crear vínculos con UUID y con label.

```
# by-label/by-uuid (filesystem properties)
ENV{DEVTYPE}=="partition", IMPORT{program}="vol_id --export $tempnode"
```

Puede conocer qué periféricos están controlados por **udev** con **udevinfo**, así como el caso de un periférico dado, de la manera siguiente:

```
# udevinfo --query=all -n /dev/sda
P: /block/sda
N: sda
S: disk/by-id/scsi-SATA_ST380011A_5JVTH798
S: disk/by-id/ata-ST380011A_5JVTH798
S: disk/by-path/pci-0000:00:14.1-scsi-0:0:0:0
S: disk/by-id/edd-int13_dev80
E: DETYPE=disk
E: ID_VENDOR=ATA
E: ID_MODEL=ST380011A
E: ID_REVISION=8.01
E: ID_SERIAL=SATA_ST380011A_5JVTH798
E: ID_SERIAL_SHORT=5JVTH798
```

El

```
E: ID_TYPE=disk
E: ID_BUS=scsi
E: ID_ATA_COMPAT=ST380011A_5JVTH798
E: ID_PATH=pci-0000:00:14.1-scsi-0:0:0:0
E: ID_EDD=int13_dev80
```

comando **udevinfo** ha sido sustituido por **udevadm info** en las últimas versiones de udev:

```
# udevadm info --query=all -n /dev/sdb
P: /devices/pci0000:00/0000:00:1f.2/host1/target1:0:0/1:0:0:0/block/sdb
N: sdb
S: disk/by-id/ata-WDC_WD5000AACS-00G8B1_WD-WCAUK0742110
S: disk/by-id/scsi-SATA_WDC_WD5000AACS-_WD-WCAUK0742110
S: disk/by-path/pci-0000:00:1f.2-scsi-1:0:0:0
S: disk/by-id/wwn-0x50014ee257f93cf8
E: UDEV_LOG=3
E: DEVPATH=/devices/pci0000:00/0000:00:1f.2/host1/target1:0:0
/1:0:0:0/block/sdb
E: MAJOR=8
E: MINOR=16
E: DEVNAME=/dev/sdb
E: DEVTYPE=disk
E: SUBSYSTEM=block
E: ID_ATA=1
E: ID_TYPE=disk
E: ID_BUS=ata
E: ID_MODEL=WDC_WD5000AACS-00G8B1
```

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

## **1. Preguntas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **La BIOS**

- 1** ¿Dónde se sitúa físicamente la BIOS?
- 2** Si ve que Linux no reconoce sus discos SATA, ¿qué debe hacer?
  - A - Sustituir los discos por SCSI o IDE.
  - B - Cambiar a una distribución más reciente.
  - C - Cambiar la compatibilidad del SATA a IDE.
  - D - Comprar una tarjeta controladora SATA compatible.
- 3** Su teclado no contesta al boot, en particular bajo GRUB. ¿Qué tiene que hacer?
- 4** ¿Quién determina el orden de detección de los discos, y por lo tanto, del boot?

### **GRUB**

- 5** ¿Cuál es el papel de GRUB?
  - A - Iniciar Linux.
  - B - Iniciar cualquier sistema operativo.
  - C - Proporcionar un modo a prueba de errores.
  - D - Visualizar un boot gráfico.
- 6** ¿Debe volver a instalar GRUB después de cada modificación?
- 7** ¿Dónde se encuentra el fichero de configuración de GRUB?
- 8** El directorio /boot que contiene el núcleo y el initrd están ubicados en una partición reconocida como la tercera del segundo disco detectado por la BIOS (hdb3, por ejemplo). ¿Cuál es la ruta al núcleo establecida dentro del fichero de configuración de GRUB?
  - A - (hd1,3)/kernel
  - B - (hd0,3)/kernel
  - C - (hd2,3)/boot/kernel
  - D - (hd1,2)/boot/kernel
- 9** No tiene tiempo para elegir una entrada del menú. ¿Cómo alargará el tiempo de espera de GRUB?
- 10** ¿Qué valor dar a «default» para iniciar por defecto en la tercera opción?
- 11** ¿Qué valor dar a la entrada «kernel» para arrancar en Windows?
- 12** ¿Qué comando debe teclear para instalar GRUB sobre el MBR del segundo disco IDE?
- 13** ¿Es posible editar las entradas de los menús directamente desde GRUB?

### **Init**

- 14** ¿Cuál es el nombre del primer proceso iniciado por el núcleo?

- A - bash
- B - initrd
- C - init
- D - kload

**15** ¿Cuál es el papel de init?

**16** ¿Para qué sirven los niveles de ejecución?

**17** Dé la ruta completa del fichero de configuración de init.

**18** ¿Qué ocurre si coloca 0 como runlevel por defecto?

**19** La línea `id:5:initdefault` fue sustituida por `default:3:initdefault:`. ¿Qué ocurre?

- A - como «default» es desconocido, init no arranca y el núcleo entra en «pánico».
- B - init pasa en runlevel 3 por defecto, ya que el identificador no significa nada.
- C - init ya no entiende la línea: pasa a modo single.
- D - se ignora la línea e init retoma la antigua configuración en runlevel 5.

**20** ¿Qué significa la línea `6:2345:respawn:/sbin/mingetty tty6`?

**21** Está en modo gráfico (nivel 5). ¿Cómo pasar a modo texto, multiusuario y con los servicios de red?

**22** ¿Qué comando permite conocer su runlevel actual?

**23** ¿Qué acción de init ejecuta un comando una sola vez al principio del boot de la máquina?

- A - sysinit
- B - bootwait
- C - once
- D - off

**24** ¿Dónde se suelen situar los servicios iniciados por el comando `rc` (o asociado)?

**25** Según los fundamentos de System V, ¿cuál es el nombre del vínculo encargado de iniciar el servicio sshd justo después del arranque de la red, sabiendo que la posición de este último es la 15<sup>a</sup>?

**26** El anterior servicio sshd se inicia en runlevels 3 y 5. Dé la lista de los directorios donde encontrar los vínculos en Debian.

**27** ¿Cómo parar el servicio ssh, sea cual sea el nivel actual, con Red Hat (hay varias respuestas posibles)?

- A - `/etc/init.d/sshd stop`
- B - `rcsshd stop`
- C - `service sshd stop`
- D - `chkconfig sshd off`

**28** En Red Hat, ¿cómo activar el servicio Apache para los niveles 3, 4 y 5?

**29** ¿Cuál es la respuesta a la pregunta anterior pero en Debian, con el orden 20?

**30** ¿Qué proceso le pide insertar su login?

**31** ¿Qué ocurre cuando da por finalizada una sesión de shell?

**32** ¿Qué comando debe insertar para programar un nuevo inicio del ordenador en 25 minutos?

## **Mensajes**

**33** ¿Cuál es el comando más importante que se debe conocer para acceder a los registros de inicio de Linux?

- A - **mensajes**
- B - **dmesg**
- C - **syslog**
- D - **cat**

**34** ¿En qué directorio se sitúan la mayoría de los registros del sistema?

**35** Un módulo llamado nvidia no se ha cargado durante el inicio del sistema. ¿Cómo saber si el núcleo ha intentado cargarlo?

### **Servicios y módulos del núcleo**

**36** ¿Se debe instalar el núcleo 2.5.63 en un servidor de producción?

**37** ¿Qué comando debe ejecutar para conocer la versión exacta de su núcleo de Linux?

- A - **uname -a**
- B - **grep kernel /var/log/messages**
- C - **uname -r**
- D - **dmesg|grep vmlinuz**

**38** `uname -r` devuelve 2.6.25.5-1.1-default. ¿Dónde se sitúan los módulos asociados?

**39** ¿Qué línea de comandos permite saber si el módulo vfat está cargado?

**40** ¿Qué nombre lleva el fichero que contiene la lista de las dependencias de los módulos y qué comando lo genera?

**41** ¿Qué línea de comandos permite recuperar la línea de descripción del módulo fan?

**42** Un módulo webcam depende de otros tres módulos. ¿Cómo cargar el módulo webcam lo más sencillamente posible?

- A - `insmod -f webcam.ko`
- B - `insmod mod1 mod2 mod3 webcam`
- C - `modprobe -r webcam`
- D - `modprobe webcam`

**43** Ya no se utiliza el módulo fat, pero aún aparece en la lista de los módulos. ¿Cómo suprimirlo?

**44** El módulo touchpad necesita el parámetro `model=12` durante su carga por el núcleo. ¿Qué línea debe usted añadir en `/etc/modprobe.conf`?

**45** Su distribución carga el módulo soundcard por defecto. Ahora bien: el módulo de su tarjeta de sonido es `snd-intel-hda`. ¿Qué debe añadir en `/etc/modprobe.conf`?

**46** ¿Qué componente cargado en el boot puede cargar los primeros módulos?

- A - `init`
- B - `sysconfig`
- C - `initrd`
- D - `vmlinuz`

**47** En Debian, ¿dónde colocar la lista de los módulos que se cargarán en el inicio?

**48** ¿Cómo conocer exactamente su modelo de procesador?

**49** Linux debe poder autorizar en total la apertura de 100000 ficheros. ¿Qué línea insertar?

**50** ¿Cómo saber si se autoriza el IP forwarding?

**51** ¿Cuál es la ruta completa del fichero que contiene los parámetros del núcleo que se debe cargar en el

boot?

## **Compilar un núcleo**

**52** ¿Dónde encontrar el último núcleo Linux oficial?

**53** ¿Cuál es la ruta por defecto de las fuentes del núcleo?

**54** ¿Es siempre útil recompilar un núcleo si se utiliza una distribución clásica?

**55** ¿En qué lenguajes se escribe Linux?

- A - en C.
- B - en ensamblador.
- C - en C++.
- D - en lenguaje máquina.

**56** ¿Qué fichero contiene las opciones de la compilación del núcleo?

**57** ¿Cuál es el mejor método para recuperar la antigua configuración si se compila un núcleo más reciente?

- A - cp /boot/config\* /usr/src/linux/.config
- B - zcat /proc/config.gz > /usr/src/linux/.config
- C - make oldconfig
- D - make cloneconfig

**58** ¿Qué comando se debe ejecutar para configurar el núcleo en modo texto?

**59** Misma pregunta, pero en modo gráfico.

**60** ¿Es útil compilar el soporte del framebuffer sobre un servidor destinado a bases de datos?

**61** ¿Qué comando permite iniciar la compilación del núcleo y de los módulos?

- A - **make**
- B - **make && make modules**
- C - **make compile\_all**
- D - **make prepare**

**62** ¿Porqué se deben instalar los módulos primero?

**63** ¿Es necesario modificar la configuración de GRUB para arrancar en el nuevo núcleo?

## **Los periféricos**

**64** ¿Cómo ve Linux los periféricos?

**65** ¿Un disco duro está en modo carácter o bloque?

**66** ¿Qué representa el número mayor?

**67** ¿Qué representa el número menor?

**68** ¿Cómo crear un periférico en modo carácter con 70 como número mayor y 2 como número menor, llamado /dev/perif1?

**69** ¿Qué comando permite listar el bus PCI?

- A - **lspci**
- B - **pcilist**
- C - **dmesg | grep pci**

- D - **pciinfo**

**70** ¿Cómo obtener los detalles del periférico USB 046d:092e?

- A - lspci -u | grep "046d:092e"
- B - lsusb -v | grep 046d:092e
- C - lssub -v -d 046d:092e
- D - cat /proc/usb/046d/092e

**71** ¿Qué driver (por lo tanto dispositivo) utiliza el IRQ 23 en su máquina?

**72** ¿Ve usted una relación entre el contenido de /proc/partitions y los ficheros periféricos asociados en /dev?

**73** ¿Cuál es la principal diferencia entre hwinfo y dmidecode?

### **USB y hotplug**

**74** Dé cuatro métodos posibles para cargar los módulos necesarios durante la carga de los periféricos USB.

**75** ¿Qué servicio gestiona el sistema de ficheros /dev en las distribuciones Linux recientes?

**76** ¿Qué hace udev cuando el núcleo indica la carga de un driver de periférico?

**77** ¿Qué hace la regla udev siguiente: KERNEL=="nvidia\*|nvidiactl\*", GROUP="video"?

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos: /77

Para este capítulo, el resultado mínimo debe ser de 58 respuestas acertadas (58/77).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- La BIOS.
- GRUB.
- Init.
- Mensajes.
- Servicios y módulos del núcleo.
- Compilar un núcleo.
- Los periféricos.
- USB y hotplug.

## **3. Respuestas**

### **La BIOS**

**1** ¿Dónde se sitúa físicamente la BIOS?

*En una EEPROM (chip) de la placa base.*

**2** Si ve que Linux no reconoce sus discos SATA, ¿qué debe hacer?

- A - Sustituir los discos por SCSI o IDE.
- B - Cambiar para una distribución más reciente.
- C - Cambiar la compatibilidad del SATA a IDE.
- D - Comprar una tarjeta controladora SATA compatible.

*C. La BIOS propone un modo Legacy IDE.*

**3** Su teclado no contesta al boot, en particular bajo GRUB. ¿Qué tiene que hacer?

*Active el modo USB legacy soporte en la BIOS.*

**4** ¿Quién determina el orden de detección de los discos y, por lo tanto, del boot?

*Le corresponde a la BIOS enumerar los discos y cargar el primer sector del primer disco.*

## **GRUB**

**5** ¿Cuál es el papel de GRUB?

- A - Iniciar Linux.
- B - Iniciar cualquier sistema operativo.
- C - Proporcionar un modo a prueba de errores.
- D - Visualizar un boot gráfico.

*B. GRUB sabe arrancar cualquier sistema, incluso Windows, e incluso desde una disquetera. También sabe proponer un boot gráfico (D).*

**6** ¿Debe volver a instalar GRUB después de cada modificación?

*No. GRUB lee el fichero de configuración, a diferencia de LILO.*

**7** ¿Dónde se encuentra el fichero de configuración de GRUB?

*En /boot/grub/menu.lst.*

**8** El directorio /boot que contiene el núcleo y el initrd están ubicados en una partición reconocida como la tercera del segundo disco detectado por la BIOS (hdb3, por ejemplo). ¿Cuál es la ruta al núcleo establecida dentro del fichero de configuración de GRUB?

- A - (hd1, 3) /kernel
- B - (hd0, 3) /kernel
- C - (hd2, 3) /boot/kernel
- D - (hd1, 2) /boot/kernel

*D. La numeración empieza a 0. hd1: segundo disco, 2: tercera partición. /boot dentro del sistema de fichero de esta partición.*

**9** No tiene tiempo para elegir una entrada del menú. ¿Cómo alargará el tiempo de espera de GRUB?

*Cambie el valor de la entrada timeout.*

**10** ¿Qué valor dar a «default» para iniciar por defecto en la tercera opción?

2.

**11** ¿Qué valor dar a la entrada «kernel» para arrancar en Windows?

*Ninguna, la entrada no debe estar presente. Debe indicar la raíz root (hdx,y) de la partición de boot Windows, y realizar un chainloader +1 para bootejar encima.*

**12** ¿Qué comando debe teclear para instalar GRUB sobre el MBR del segundo disco IDE?

`grub-install /dev/hdb`

**13** ¿Es posible editar las entradas de los menús directamente desde GRUB?

*Sí. GRUB propone un shell especializado en el arranque. Se pueden modificar las opciones en línea.*

### Init

**14** ¿Cuál es el nombre del primer proceso lanzado por el núcleo?

- A - bash
- B - initrd
- C - init
- D - kload

*C. init.*

**15** ¿Cuál es el papel de init?

*Inicia y para los servicios del sistema.*

**16** ¿Para qué sirven los niveles de ejecución?

*Para definir varios estados o configuraciones de Linux y poder pasar de un estado a otro fácilmente.*

**17** Dé la ruta completa del fichero de configuración de init.

`/etc/inittab`

**18** ¿Qué ocurre si coloca 0 como runlevel por defecto?

*En cuanto el núcleo ejecute init, éste apagará la máquina.*

**19** La línea `id:5:initdefault:` fue sustituida por `default:3:initdefault:` ¿Qué ocurre?

- A - como «default» es desconocido, init no arranca y el núcleo entra en «pánico».
- B - init pasa en runlevel 3 por defecto, ya que el identificador no significa nada.
- C - init ya no entiende la línea: pasa a modo single.
- D - se ignora la línea e init retoma la antigua configuración en runlevel 5.

*B. El id está aquí sólo para usted e init no lo interpreta. El nivel 3 es el nuevo nivel por defecto.*

**20** ¿Qué significa la línea `6:2345:respawn:/sbin/mingetty tty6`?

*Se ejecutará el comando mingetty en los niveles 2, 3, 4 y 5, con el terminal virtual 6 (accesible vía Alt-F6) como valor de parámetro. Si se quita o mata, respawn indica a init que debe volver a ejecutarlo automáticamente.*

**21** Está en modo gráfico (nivel 5). ¿Cómo pasar a modo texto, multiusuario y con los servicios de red?

*Es el nivel 3. Teclee telinit 3 o init 3.*

**22** ¿Qué comando permite conocer su runlevel actual?

**runlevel**

**23** ¿Qué acción de init ejecuta un comando una sola vez al principio del boot de la máquina?

- A - sysinit
- B - bootwait
- C - once
- D - off

*A. Se ejecuta el comando relacionado con la acción sysinit al principio del arranque del sistema.*

**24** ¿Dónde se suelen situar los servicios iniciados por el comando **rc** (o asociado)?

*En /etc/init.d.*

**25** Según los fundamentos de System V, ¿cuál es el nombre del vínculo encargado de iniciar el servicio sshd justo después del arranque de la red, sabiendo que la posición de este último es la 15<sup>a</sup>?

*S16sshd*

**26** El anterior servicio sshd se inicia en runlevels 3 y 5. Dé la lista de los directorios donde encontrar los vínculos en Debian.

*/etc/rc3.d y /etc/rc5.d.*

**27** ¿Cómo parar el servicio ssh, sea cual sea el nivel actual, con Red Hat (hay varias respuestas posibles)?

- A - /etc/init.d/sshd stop
- B - rcsshd stop
- C - service sshd stop
- D - chkconfig sshd off

*A y C. B funciona en openSUSE. D suprime el servicio de todos los runlevels, pero no lo para si está iniciado actualmente.*

**28** En Red Hat, ¿cómo activar el servicio Apache para los niveles 3, 4 y 5?

*chkconfig --level 345 apache on*

**29** ¿Cuál es la respuesta a la pregunta anterior pero en Debian, con el orden 20?

*update-rc.d apache start 20 3 4 5 . stop 10 0 1 6, por ejemplo.*

**30** ¿Qué proceso le pide insertar su login?

*El emulador de terminal iniciado por init.*

**31** ¿Qué ocurre cuando da por finalizada una sesión de shell?

*init vuelve a iniciar un terminal.*

**32** ¿Qué comando debe insertar para programar un nuevo inicio del ordenador en 25 minutos?

*shutdown -r +25*

## **Mensajes**

**33** ¿Cuál es el comando más importante que se debe conocer para acceder a los registros de inicio de Linux?

- A - **mensajes**
- B - **dmesg**
- C - **syslog**
- D - **cat**

*B. Es el comando más utilizado en particular para detectar los problemas de inicio y detección del hardware.*

**34** ¿En qué directorio se sitúan la mayoría de los registros del sistema?

*En /var/log.*

**35** Un módulo llamado nvidia no se ha cargado durante el inicio del sistema. ¿Cómo saber si el núcleo ha intentado cargarlo?

*grep nvidia /var/log/messages*

## **Servicios y módulos del núcleo**

**36** ¿Se debe instalar el núcleo 2.5.63 en un servidor de producción?

*No. La segunda cifra es impar: es una rama de desarrollo.*

**37** ¿Qué comando debe ejecutar para conocer la versión exacta de su núcleo de Linux?

- A - **uname -a**
- B - **grep kernel /var/log/messages**
- C - **uname -r**
- D - **dmesg|grep vmlinuz**

*C. La -r significa release.*

**38** `uname -r` devuelve 2.6.25.5-1.1-default. ¿Dónde se sitúan los módulos asociados?

*En /lib/modules/2.6.25.5-1.1-default.*

**39** ¿Qué línea de comandos permite saber si el módulo vfat está cargado?

*lsmod|grep vfat*

**40** ¿Qué nombre lleva el fichero que contiene la lista de las dependencias de los módulos y qué comando lo genera?

*depmod genera modules.dep.*

**41** ¿Qué línea de comandos permite recuperar la línea de descripción del módulo fan?

*modinfo fan|grep ^description. Devuelve: «Descripción: ACPI Fan Driver».*

**42** Un módulo webcam depende de otros tres módulos. ¿Cómo cargar el módulo webcam lo más sencillamente posible?

- A - `insmod -f webcam.ko`

- B - insmod mod1 mod2 mod3 webcam
- C - modprobe -r webcam
- D - modprobe webcam

*D. Si las dependencias son correctas, modprobe cargará todos los módulos asociados.*

**43** Ya no se utiliza el módulo fat, pero aún aparece en la lista de los módulos. ¿Cómo suprimirlo?

*rmmod fat*

**44** El módulo touchpad necesita el parámetro model=12 durante su carga por el núcleo. ¿Qué línea debe usted añadir en /etc/modprobe.conf?

*options touchpad model=12*

**45** Su distribución carga el módulo soundcard por defecto. Ahora bien: el módulo de su tarjeta de sonido es snd-intel-hda. ¿Qué debe añadir en /etc/modprobe.conf?

*alias soundcard snd-intel-hda*

**46** ¿Qué componente cargado en el boot puede cargar los primeros módulos?

- A - init
- B - sysconfig
- C - initrd
- D - vmlinuz

*C. El ramdisk inicial cargado por el núcleo según la configuración de GRUB contiene comandos y scripts que cargan los módulos necesarios, como el soporte de los controladores de discos y sistemas de ficheros.*

**47** En Debian, ¿dónde colocar la lista de los módulos que se cargarán en el inicio?

*En /etc/modules.*

**48** ¿Cómo conocer exactamente su modelo de procesador?

*Por /proc/cpuinfo.*

**49** Linux debe poder autorizar en total la apertura de 100000 ficheros. ¿Qué línea insertar?

*echo 100000 > /proc/sys/fs/file-max*

**50** ¿Cómo saber si se autoriza el IP forwarding?

*sysctl -a | grep ip\_forward. Si el valor es a 1, se autoriza el forwarding.*

**51** ¿Cuál es la ruta completa del fichero que contiene los parámetros del núcleo que se debe cargar en el boot?

*/etc/sysctl.conf*

### **Compilar un núcleo**

**52** ¿Dónde encontrar el último núcleo Linux oficial?

*En el sitio kernel.org.*

**53** ¿Cuál es la ruta por defecto de las fuentes del núcleo?

*/usr/src/linux.*

**54** ¿Es siempre útil recompilar un núcleo si se utiliza una distribución clásica?

No. El editor suele adaptar o parchear los núcleos de las distribuciones Linux. Verifique si éste propone actualizaciones antes de compilar su propio núcleo, con el consiguiente riesgo de perder el soporte.

**55** ¿En qué lenguajes se escribe Linux?

- A - en C.
- B - en ensamblador.
- C - en C++.
- D - en lenguaje máquina.

A y B. Mayoritariamente en C y el ensamblador, según las diferentes arquitecturas.

**56** ¿Qué fichero contiene las opciones de la compilación del núcleo?

.config.

**57** ¿Cuál es el mejor método para recuperar la antigua configuración si se compila un núcleo más reciente?

- A - cp /boot/config\* /usr/src/linux/.config
- B - zcat /proc/config.gz > /usr/src/linux/.config
- C - make oldconfig
- D - make cloneconfig

C. Se le pedirá luego la configuración que le falta.

**58** ¿Qué comando se debe ejecutar para configurar el núcleo en modo texto?

make menuconfig

**59** Misma pregunta, pero en modo gráfico.

make xconfig o make gconfig

**60** ¿Es útil compilar el soporte del framebuffer sobre un servidor destinado a las bases de datos?

No. Es verdad que quien puede más puede menos, pero ¿por qué sobrecargar el núcleo de manera inútil?

**61** ¿Qué comando permite iniciar la compilación del núcleo y de los módulos?

- A - **make**
- B - **make && make modules**
- C - **make compile\_all**
- D - **make prepare**

A. Un simple make basta: es la acción por defecto.

**62** ¿Por qué se deben instalar los módulos primero?

El proceso de instalación necesita la presencia de módulos y ficheros de dependencias, en particular para construir el initrd.

**63** ¿Es necesario modificar la configuración de GRUB para arrancar en el nuevo núcleo?

*Eso depende de las distribuciones, pero en principio el make install añade las entradas que faltan en el fichero de configuración. No perderá su tiempo si lo comprueba.*

## **Los periféricos**

**64** ¿Cómo ve Linux los periféricos?

*Como ficheros.*

**65** ¿Un disco duro está en modo carácter o bloque?

*En modo bloque.*

**66** ¿Qué representa el número mayor?

*Identifica el driver, o el controlador o ambos.*

**67** ¿Qué representa el número menor?

*El propio periférico, o el número de periférico sobre el controlador o una particularidad del periférico...*

**68** ¿Cómo crear un periférico en modo carácter con 70 como número mayor y 2 como número menor, llamado /dev/perif1?

*mknod /dev/perif1 c 70 2*

**69** ¿Qué comando permite listar el bus PCI?

- A - **lspci**
- B - **pcilist**
- C - **dmesg|grep pci**
- D - **pciinfo**

*A. El comando lspci lista todos los adaptadores PCI.*

**70** ¿Cómo obtener los detalles del periférico USB 046d:092e?

- A - **lspci -u | grep "046d:092e"**
- B - **lsusb -v|grep 046d:092e**
- C - **lssub -v -d 046d:092e**
- D - **cat /proc/usb/046d/092e**

*C.*

**71** ¿Qué driver (por lo tanto dispositivo) utiliza el IRQ 23 en su máquina?

*grep "23:" /proc/interrupts*

**72** ¿Ve usted una relación entre el contenido de /proc/partitions y los ficheros periféricos asociados en /dev?

*Sí, los números mayores y menores están relacionados, y la terminación numérica del fichero tiene una relación con el número menor. Por ejemplo, sda1 (8,1) y sdb1 (8,16+1, ya que sólo puede haber 15 particiones además del propio disco...).*

**73** ¿Cuál es la principal diferencia entre hwinfo y dmidecode?

*hwinfo consulta a los dispositivos, concretamente a través de los drivers, mientras que dmidecode se dirige a*

*la tabla física DMI instalada por la SMBIOS y la placa base. Los resultados pueden ser diferentes.*

## **USB y hotplug**

**74** Dé cuatro métodos posibles para cargar los módulos necesarios durante la carga de los periféricos USB.

*1- initrd durante la carga del núcleo. 2- Un servicio iniciado muy temprano por init. 3- El Kernel Module Loader, que detecta y carga los módulos USB en función de los identificadores físicos. 4- udev o hotplug, que puede cargar módulos complementarios y crear los periféricos asociados.*

**75** ¿Qué servicio gestiona el sistema de ficheros /dev en las distribuciones Linux recientes?

*Es udev, que está a la escucha de los eventos del núcleo.*

**76** ¿Qué hace udev cuando el núcleo indica la carga de un driver de periférico?

*Busca en sus reglas una correspondencia con el evento recibido y aplica las consignas asociadas si encuentra algunas.*

**77** ¿Qué hace la regla udev siguiente: KERNEL=="nvidia\*|nvidiactl\*", GROUP="video"?

*Cuando recibe un evento del núcleo que empieza por nvidia o nvidiactl, cambia el grupo del periférico asociado a «video».*

# Requisitos y objetivos

## 1. Requisitos

- Disponer de privilegios de root.
- Saber utilizar el shell y los comandos.
- Saber utilizar un editor.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Comprender las nociones de gestión de los usuarios.
- Crear y modificar usuarios y grupos.
- Implementar una política de contraseñas.
- Conocer los comandos avanzados.
- Publicar las notificaciones a los usuarios.
- Hacerse una idea de los PAM.
- Gestionar las impresoras.
- Automatizar tareas.
- Utilizar algunas herramientas de backup.
- Poner el sistema en hora.

# Prácticas

## 1. Gestión de los usuarios

Objetivo: crear un usuario y aplicar una política de seguridad.

1. Cree un usuario tom que tenga el UID 1200, el grupo users, el comentario «Cazador de Jerry» y el shell /bin/bash. Por supuesto, cree el usuario con su directorio personal:

```
# useradd -m -u 1200 -g users -c "Cazador de Jerry" tom
```

2. Dé la contraseña «tomcat» a tom. Veamos un truco: usted puede crear una contraseña sin inserción mediante el comando **passwd** con el parámetro **--stdin**:

```
# echo tomcat | passwd --stdin tom
```

3. Añada un grupo cat con el GID 530:

```
# groupadd -g 530 cat
```

4. Añada a tom en el grupo cat editando el fichero **/etc/group**. Para ello, añada el nombre tom al final de la línea correspondiente.

```
cat:x:530:tom
```

5. Modifique la información de cambio de contraseña de tom con el comando **chage**. No se puede cambiar la contraseña antes de 10 días, y es obligatorio cambiarla cada 50 días.

```
# chage tom
Changing agin information for tom
    Minimum Password Age [7] : 10
    Maximum Password Age [40] : 50
    ...
...
```

6. Suprime al usuario tom. Ejecute luego el comando **pwck**. Rectifique el problema.

```
# userdel tom
# pwck
```

El comando le informa de que el grupo cat contiene un usuario tom que ya no existe. Por lo tanto, debe quitar a tom del grupo cat para conservar la coherencia del sistema.

7. Avise a sus usuarios de que se parará el sistema el 12 de diciembre a las 17:00 por mantenimiento. Puede utilizar el fichero **/etc/issue**, el fichero **/etc/motd** y el fichero **/etc/issue.net** para escribir su texto.

## 2. La impresión

Objetivo: configurar las impresoras e imprimir ficheros.

1. Puede elegir entre tres sistemas de impresión: BSD, System V y CUPS. ¿Qué sistema debería utilizar?  
CUPS, ya que es compatible con los dos primeros.

2. Conéctese a la interfaz de administración de CUPS mediante un navegador Web:  
`http://localhost:631`
3. En una de las impresoras configuradas, verifique el estado de las impresiones e imprima una página de prueba.
4. ¿Qué ocurre si intenta imprimir una imagen png así: `lpr fichero.png`?  
Funciona perfectamente: el sistema de impresión CUPS dispone de filtros que gestionan la mayoría de los documentos transformándolos en el lenguaje de impresión de la impresora y en particular en PostScript.
5. Anule todas las impresiones en curso:

```
# lprm -
```

### 3. Automatización de tareas

Objetivo: utilizar el crontab.

1. Ejecute el comando `ps -ef` para el usuario root cada 5 minutos y redireccione el resultado en `/tmp/ps_result`, sin chafar los antiguos:

```
# crontab -e root
```

Añada la línea siguiente:

```
*/5 * * * * ps -ef >> /tmp/ps_result
```

2. Verifique la lista de las tareas en crontab:

```
# crontab -l
```

3. Espere cinco minutos y verifique el resultado en `/tmp`.
4. Cron interpreta todos los ficheros colocados en `/etc/cron.d` de manera automática. Por lo tanto, resulta inútil modificar el crontab del usuario root si el comando es permanente: añada un fichero `/etc/cron.d/cron_ps` con la línea anterior dentro.
5. Prohiba al usuario joe utilizar el contrab. Modificar el fichero `/etc/cron.deny` y añada al usuario joe en una nueva línea de este fichero.

### 4. Las trazas del sistema

Objetivo: estudiar las trazas del sistema y la configuración de syslog.

1. Las trazas del inicio del sistema y su principal información son accesibles desde el comando `dmesg`. Aíslle la línea que indica los parámetros de boot del núcleo:

```
# dmesg | grep -i command
```

Encontrará esta línea en la configuración de GRUB.

2. El comando `last` devuelve también la fecha de la última parada y el último reinicio del sistema. Recupere la lista de los últimos reboots:

```
# last | grep reboot
```

Aunque esto funcione, lo mejor sigue siendo:

- ```
# last reboot
```
3. El fichero **/var/log/message** contiene las trazas de los eventos importantes del sistema, incluso los anteriores al boot. Puede haber varios por culpa de logrotate. En el último, ¿cuál es la información relativa a eth0?
- ```
# dmesg | grep eth0
```
4. Va a colocar todas las trazas correspondientes al núcleo en un fichero llamado kernel.log. Modifique el fichero /etc/syslog.conf en consecuencia. Añada la línea:
- ```
kern.* /var/log/kernel.log
```
5. Fuerce syslog a reinicializarse y volver a leer su configuración. Si su distribución no propone un método que utilice los servicios, entonces mande la señal SIGHUP al proceso de syslog:

```
# ps -e | grep syslog| awk '{print $1}'  
4577  
# pkill syslog  
4577  
# kill -HUP 4577
```

## 5. Archivado

Objetivo: aprender a crear y a manejar los archivos de ficheros.

1. El comando **gzip** sirve para comprimir un fichero. Sin embargo, su funcionamiento puede sorprender un poco. Cree un fichero de texto ordinario y comprímalo:

```
# gzip fic.txt
```

El archivo resultante es fic.txt.gz. Pero ¿dónde está el fichero de origen? Fue suprimido.

2. Para evitar este problema, tiene una solución: comprimir el fichero y forzar al resultado a pasar por la salida estándar. Redirija la salida a un fichero.

```
# gzip -c fic.txt >fic.txt.gz
```

3. Efectúe una copia de seguridad en un archivo comprimido al formato gzip de su directorio personal. Utilice únicamente el comando **tar** para ello:

```
# tar cvzf home.tgz /home
```

4. Descomprima este archivo en el lugar de su elección, pero esta vez utilizando tar, gzip y una tubería:

```
# gzip -cd home.tgz | tar xvf -
```

# Administración de los usuarios

## 1. Fundamentos

### a. Identificación y autenticación

La **identificación** consiste en saber quién es quién para determinar los permisos de la persona que se conecta. Se identifica un usuario mediante un login.

La **autenticación** consiste en aportar la prueba de quiénes somos mediante, por ejemplo, un secreto compartido entre el usuario y el sistema, y que sólo conocen ellos. Se autentifica el usuario con una contraseña.

### b. Los usuarios

Un usuario es la asociación de un nombre de conexión -el login- con un UID y al menos un GID.

- **UID:** User ID.
- **GID:** Group ID.

Los UID y los GID suelen ser únicos. El login es único. Sin embargo, se puede considerar la asociación de varios logins al mismo UID, sabiendo que el sistema trabaja a veces con el login.

El UID identifica el usuario (o la cuenta asociada) a lo largo de su conexión. Se utiliza para el control de sus derechos y de los de los procesos que ha iniciado. Lo que se almacena en la tabla de los inodos, dentro de la tabla de los procesos, son los UID y GID, y no los logins.

El usuario dispone de los atributos básicos siguientes:

- un nombre de conexión llamado «login»;
- una contraseña;
- un UID;
- un GID correspondiente a su grupo principal;
- una descripción;
- un directorio de conexión;
- un comando de conexión.

Hay otros atributos disponibles mediante la utilización de la seguridad de las contraseñas en el fichero shadow (ver la sección correspondiente).

Se suelen asociar los UID de un valor inferior a 100 a cuentas especiales con derechos extendidos. Así el UID de root, el administrador, es 0. Según las distribuciones, a partir de 100, 500 o 1000, y hasta 65.535 ( $2^{16}-1$ ) aproximadamente son los UID de los usuarios sin privilegios particulares. Estos parámetros pueden modificarse en `etc/login.defs`.

Un login tiene en principio un tamaño de 8 caracteres. En realidad, Linux y otros sistemas aceptan un tamaño mayor, pero con la mayoría de los comandos la visualización de los logins, incluso su gestión, está limitada a 8 caracteres.

Un login acepta la mayoría de los caracteres. No debe empezar por una cifra. Es posible modificar la lista de los caracteres autorizados y forzar la longitud y la complejidad mediante los mecanismos de autenticación

PAM y el fichero `/etc/login.defs`.

### c. Los grupos

Cada usuario forma parte de al menos un grupo. Un grupo agrupa usuarios. Como para los logins, el GID del grupo siempre acompaña al usuario para controlar sus privilegios. Un usuario puede formar parte de varios grupos. En este caso, hay que distinguir su grupo primario de los grupos secundarios.

Los grupos son también números (GID). Existen grupos específicos para la gestión de algunas propiedades del sistema y, en particular, el acceso a ciertos periféricos.

Cada vez que un usuario crea un fichero, éste recibe como valor de grupo con privilegios el primario del creador. Si el usuario seb tiene como grupo primario **users**, entonces los ficheros creados por seb tendrán como grupo con privilegios **users**.

Un usuario dispone de todos los derechos asociados a sus grupos secundarios. Si seb tiene como grupo secundario **video** y un fichero dispone de los derechos de escritura para este grupo, entonces seb tendrá derecho a modificar su contenido.

El comando **id** permite conocer la información esencial sobre un usuario: uid, gid, grupos secundarios.

```
$ id seb  
uid=1000(seb) gid=100(users) grupos=100(users),16(dialout),3(sys),33(video)
```

Seb ha creado un fichero. Su propietario es seb y su grupo es el grupo primario de seb: users.

```
$ touch test  
$ ls -l test  
-rw-r--r-- 1 seb users 0 abr 10 14:30 test
```

### d. Las contraseñas

Las contraseñas permiten autenticar a los usuarios. Deben ser lo bastante complejas como para que no se puedan descubrir fácilmente, pero lo bastante intuitivas como para que se pueda recordar. Las contraseñas están cifradas (MD5, SHA, Blowfish, por ejemplo) y no son directamente legibles bajo su forma cifrada por el usuario con el fin de que nadie pueda intentar descifrarlas.

Un usuario debería cambiar regularmente su contraseña, no escribirla nunca en ninguna parte ni conservarla encima. Luego veremos cómo es posible obligar a un usuario a aplicar reglas de creación y duración de contraseñas.

Veamos por ejemplo el resultado encriptado por Blowfish (reconocible por el `$2a$` que empieza la cadena) de una contraseña:

```
aher874oP47 vale  
$2a$10$CqutecUALTGSFs2BPnVl..ntI80Edy5j6gLI/cIKhHP4XZISdLGZ0
```

## 2. Los ficheros

### a. /etc/passwd

El fichero `/etc/passwd` contiene la lista de los usuarios del sistema local. Cualquier usuario puede leerlo. La información que contiene es pública y útil tanto para el sistema como para los usuarios. Cada línea representa un usuario y se compone de siete campos.

Login:password:UID:GID:comment:homedir:shell

- Campo 1: el login o nombre de usuario.
- Campo 2: en las antiguas versiones, la contraseña cifrada. Si hay una x, se coloca la contraseña en `/etc/shadow`. Si es un signo de exclamación, se bloquea la cuenta.
- Campo 3: el User ID.
- Campo 4: el GID, o sea, el grupo principal.
- Campo 5: un comentario o descripción. Es un campo de información.
- Campo 6: el directorio de trabajo, personal, del usuario. Es el directorio al que llega cuando se conecta.
- Campo 7: el shell por defecto del usuario. Pero puede ser cualquier otro comando, incluso un comando que prohíbe la conexión.

## b. `/etc/group`

El fichero `/etc/group` contiene la definición de los grupos de usuarios, y para cada uno la lista de los usuarios de los cuales es el grupo secundario. Cada línea se compone de cuatro campos:

`Group:password:GID:user1,user2...`

- Campo 1: el nombre del grupo.
- Campo 2: la contraseña asociada. Veremos la explicación justo a continuación.
- Campo 3: el Group Id.
- Campo 4: la lista de los usuarios que forman parte de este grupo.

Es inútil colocar en el cuarto campo a los usuarios que tienen este grupo como grupo principal; lo maneja el sistema.

Le puede sorprender la presencia de un campo de contraseña para los grupos. En la práctica se utiliza pocas veces. Como por supuesto es imposible conectarse como grupo, la explicación está en otra parte. Un usuario tiene derecho a cambiar de grupo para coger, de manera temporal al menos, un grupo secundario como grupo primario con el comando **newgrp**.

En este caso, el administrador puede implementar una contraseña sobre el grupo para proteger el acceso a este grupo como grupo principal.

## c. `/etc/shadow`

El fichero `/etc/shadow` acompaña al fichero `/etc/passwd`. Se coloca en él, entre otras cosas, la contraseña cifrada de los usuarios. Para ser más precisos, contiene toda la información sobre la contraseña y su validez en el tiempo. Cada línea se compone de 9 campos separados por «::»:

`bean:$2a$10$AjADxPEfE5iUJcltzYA4wOZ0.f2UZ0qP/8En0FY.P.m10HifS7J8i:13913:0:99999:7:::`

- Campo 1: el login.
- Campo 2: la contraseña cifrada. El `$xx$` inicial indica el tipo de cifrado.
- Campo 3: número de días desde el 1º de enero de 1970 al último cambio de contraseña.
- Campo 4: número de días sin poder cambiar la contraseña (0: se puede cambiar en cualquier momento).

- Campo 5: número de días a partir de los cuales se debe cambiar la contraseña.
- Campo 6: número de días antes del vencimiento de la contraseña durante los cuales se debe avisar al usuario.
- Campo 7: número de días después del vencimiento de la contraseña tras los cuales se desactiva la cuenta.
- Campo 8: número de días desde el 1º de enero de 1970 hasta el momento en el cual se desactivó la cuenta.
- Campo 9: reservado.

En el ejemplo de la línea bean, se ha cambiado la contraseña 13913 días después del 01/01/1970. Se debe cambiar la contraseña antes de 0 días, pero seguirá siendo válida (la contraseña), ya que el campo siguiente indica que hay que cambiarla después de 99999 días (273 años) y el campo 5 está vacío (sin obligación de cambio de contraseña). Se desactiva la cuenta después de 7 días: no hay riesgo de que ocurra...

Los valores actuales para el cifrado de contraseñas son los siguientes:

- **\$1\$**: MD5
- **\$2a\$**: Blowfish
- **\$5**: SHA-256
- **\$6**: SHA-512
- Otro: DES

 Para conocer la fecha en función del 01/01/1970, utilice el comando date como a continuación al añadir el número de días deseado:

```
$ date --date "1 jan 1970 +13984 days"
mar abr 15 00:00:00 CEST 2008
```

#### d. /etc/gshadow

El fichero `/etc/gshadow` es el equivalente del fichero anterior, pero para los grupos. Sin embargo, la mayoría de las distribuciones Linux no lo soportan por defecto, así que no se explicará aquí. Se colocan las contraseñas de los grupos en `/etc/group`.

### 3. Gestión de los usuarios

#### a. Creación

La creación de un usuario podría ser totalmente manual, ya que Linux (y los demás Unix) se apoya en una serie de comandos que «sólo» modifican ficheros planos ya existentes y que crean y vuelven a copiar ficheros y carpetas en el sitio correcto con los privilegios correctos.

La creación de un usuario consiste en:

- añadir una línea en `/etc/passwd`,
- añadir una línea en `/etc/shadow`,

- añadir una información si es preciso en `/etc/group`,
- crear el directorio personal y actualizar su contenido con `/etc/skel`,
- cambiar los permisos y el propietario del directorio personal,
- cambiar la contraseña (cifrada).

Usted puede crear directamente una cuenta editando los ficheros con un editor, aunque esto no es nada aconsejable. Si desea hacerlo de todas maneras, utilice el comando **vipw**, que actualizará las diversas cachés asociadas a la gestión de las cuentas.

El comando **vipw** admite tres argumentos:

- `-p`: edición de `/etc/passwd`.
- `-g`: edición de `/etc/group`.
- `-s`: edición de `/etc/shadow`.

Evidentemente, en la práctica hay que evitar editar estos archivos a mano, ya que hay muchos otros mecanismos creados a partir del uso de comandos específicos para la creación o modificación de cuentas, como por ejemplo la comprobación de la complejidad de la contraseña.

Cabe destacar que existe una restricción asociada al formato del login. Responde a una norma POSIX: sólo puede empezar por un carácter alfabético, seguido de caracteres portables (letras, cifras, guión, guión bajo, etc.).

Todo esto se puede llevar a cabo con el comando **useradd**. Añade una nueva cuenta y efectúa las principales operaciones:

- creación del usuario y edición de los ficheros,
- creación de un grupo privado del usuario (con el mismo nombre que éste),
- creación del directorio personal, edición y modificación de los derechos.

`useradd <options> login`

Si no se ha especificado ninguna opción, se recuperarán los valores por defecto dentro del fichero `/etc/defaults/useradd`. Se aceptan las opciones principales siguientes:

| Opción    | Papel                                                                                                                                                                                                  |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-m</b> | Crea también el directorio personal. Se incluye a veces por defecto, pero es mejor comprobar que el directorio personal esté presente después de la utilización del comando si no utiliza esta opción. |
| <b>-u</b> | Especifica el UDI numérico del usuario, para forzarlo. Dicho de otro modo, se calcula el UID según las reglas del fichero <code>login.defs</code> y los UID existentes.                                |
| <b>-g</b> | Especifica el grupo primario del usuario, por GID o por su nombre (variable GROUP).                                                                                                                    |
| <b>-G</b> | Especifica los grupos adicionales (secundarios, del usuario) separados por comas (variable GROUPS).                                                                                                    |
| <b>-d</b> | Ruta del directorio personal. En general <code>/home/&lt;login&gt;</code> , pero no se puede                                                                                                           |

|           |                                                                                                                                                                                                     |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|           | especificar cualquier ruta (variable HOME/<login>).                                                                                                                                                 |
| <b>-c</b> | Un comentario asociado a la cuenta. Puede ser cualquiera, pero a veces algunos comandos como <b>finger</b> lo pueden utilizar. El usuario puede modificar su contenido con el comando <b>chfn</b> . |
| <b>-k</b> | Ruta del directorio que contiene el esqueleto del árbol del directorio usuario. Suele ser /etc/skel (variable SKEL).                                                                                |
| <b>-s</b> | Shell (intérprete de comandos) por defecto del usuario (variable SHELL). El usuario puede cambiarlo mediante el comando <b>chsh</b> .                                                               |
| <b>-p</b> | La contraseña del usuario. ¡Cuidado! ¡La contraseña debe estar ya cifrada! A no ser que se vuelva a copiar la contraseña de una cuenta genérica, lo mejor es usar el comando <b>passwd</b> .        |

El comando siguiente crea la cuenta roberto con la mayoría de las opciones básicas especificadas. Es sólo un ejemplo, excepto a veces la **-m**, ya que si usted no especifica nada son las opciones por defecto en relación con las especificadas en el fichero **/etc/defaults/useradd**.

```
# useradd -m -u 1010 -g users -G video,dialout,lp -s /bin/bash -d
/home/roberto -c "Cuenta de Roberto" roberto
# grep roberto /etc/passwd
robert:x:1010:100:Cuenta de Roberto:/home/roberto:/bin/bash
```

comando no crea la contraseña. Hay que hacerlo a mano con el comando **passwd**.

```
# passwd roberto
Changing password for roberto.
Nueva contraseña:
Vuelva a introducir la nueva contraseña:
Contraseña cambiada.
```

## b. Seguridad de las contraseñas

### Cambiar la contraseña

El comando **passwd** permite gestionar las contraseñas, pero también las autorizaciones de conexión, así como la mayoría de los campos presentes en **/etc/shadow**.

Cualquier usuario tiene derecho a cambiar su contraseña en el plazo especificado por el campo 4 de **/etc/shadow**. La acción por defecto consiste en cambiar la contraseña del usuario corriente. Se requiere la antigua contraseña por seguridad (en particular para evitar que una persona mal intencionada modifique su contraseña a sus espaldas). La inserción está enmascarada.

```
$ id
uid=1000(seb) gid=100(users) ...
$ passwd
Changing password for seb.
Antigua contraseña:
Nueva contraseña:
Vuelva a introducir la nueva contraseña:
Contraseña cambiada.
```

Los módulos **PAM** (*Pluggable Authentication Module*) pueden imponer exigencias más o menos estrictas para el cambio de la contraseña: en cuanto a la longitud, que no se base en una palabra del diccionario, etc.

Veamos lo que ocurre cuando se quiere utilizar pepe (demasiado corto), qwerty (demasiado largo) y María (diccionario):

```
$ passwd  
Changing password for seb.  
Antigua contraseña:  
Nueva contraseña:  
Contraseña incorrecta: demasiado corta  
Nueva contraseña:  
Contraseña incorrecta: demasiado simple  
Nueva contraseña:  
Contraseña incorrecta: basada en una palabra del diccionario  
passwd: Número máximo de intentos agotado para el servicio
```

El usuario root tiene derecho a modificar las contraseñas de todos los usuarios del sistema, sin que sea preciso que conozca la contraseña anterior. Aún mejor: puede forzar el uso de una contraseña incluso aunque no haya sido validada por PAM:

```
# passwd seb  
Changing password for seb.  
Nueva contraseña:  
Contraseña incorrecta: basada en una palabra del diccionario  
Vuelva a introducir la nueva contraseña:  
Contraseña cambiada.
```

### **Gestionar la validez**

Se pueden modificar todos los campos de `/etc/shadow` con el comando **passwd**. Veamos algunas de las opciones disponibles.

| Opción              | Papel                                                                                                                  |
|---------------------|------------------------------------------------------------------------------------------------------------------------|
| <b>-l</b>           | Lock: bloquea una cuenta al añadir un ! delante de la contraseña cifrada.                                              |
| <b>-u</b>           | Unlock: desbloquea la cuenta. No es posible activar una cuenta que no tenga contraseña. Hay que utilizar -f para ello. |
| <b>-d</b>           | (root) Suprime la contraseña de la cuenta.                                                                             |
| <b>-n &lt;j&gt;</b> | (root) Duración de vida mínima en días de la contraseña.                                                               |
| <b>-x &lt;j&gt;</b> | (root) Duración de vida máxima en días de la contraseña.                                                               |
| <b>-w &lt;j&gt;</b> | (root) Número de días antes de un aviso.                                                                               |
| <b>-i &lt;j&gt;</b> | (root) Período de gracia antes de la desactivación si ha vencido la contraseña.                                        |
| <b>-S</b>           | (root) Estatuto de la cuenta.                                                                                          |

En el ejemplo siguiente, se ha modificado la cuenta bean de esta manera:

- Debe esperar 5 días después de la inserción de una nueva contraseña para poder cambiarla.
- Su contraseña es válida 45 días.

- Se le avisa 7 días antes de que deba cambiar la contraseña.
- Si no cambia la contraseña después de 45 días, dispone todavía de 5 días antes de que sea bloqueada.

```
# passwd -n 5 -x 45 -w 7 -i 5 bean
Password expiry information changed.
```

Veamos la línea de **/etc/shadow** asociada.

```
El
bean:$2a$10$dwbUGrC75bs3l52V5DHxZefkZyB6VTHsLH5ndjsNe/vF/HAzH0cR2:13984:5:45:7:5::
```

comando **chage** permite hacer más o menos lo mismo. Es un comando del root. Iniciado sin otro argumento que el login del usuario, es interactivo. Observe al final la posibilidad de modificar la fecha del último cambio de la contraseña y una fecha fija de expiración de la contraseña (campo 8):

```
# chage bean
Changing aging information for bean.
      Minimum Password Age [7]:
      Maximum Password Age [40]:
      Password Expiration Warning [10]:
      Password Inactive [5]:
      Last Password Change (YYYY-MM-DD) [2008-04-10]:
      Account Expiration Date (YYYY-MM-DD) [1969-12-31]: 2010-01-01
Aging information changed.
```

Veamos la línea **/etc/shadow** resultante:

```
bean:$2a$10$dwbUGrC75bs3l52V5DHxZefkZyB6VTHsLH5ndjsNe/vF/HAzH0cR2:13979:7:40:10:5:14610:
```

Se aceptan los parámetros siguientes:

| Opción    | Papel                                                                |
|-----------|----------------------------------------------------------------------|
| <b>-m</b> | Mindays: equivale a passwd -n.                                       |
| <b>-M</b> | Maxdays: equivale a passwd -x.                                       |
| <b>-d</b> | Fecha de última modificación de la contraseña (desde el 01/01/1970). |
| <b>-E</b> | Fecha de vencimiento de la contraseña (desde el 01/01/1970).         |
| <b>-I</b> | Inactive: equivale a passwd -i.                                      |
| <b>-W</b> | Warndays: equivale a passwd -w.                                      |
| <b>-l</b> | List: muestra todos los detalles.                                    |

Los detalles son mucho más legibles con **chage** que con **passwd**:

```
# passwd -S bean
bean PS 04/10/2008 7 40 10 5
```

```
# chage -l bean
Minimum:          7
Maximum:         40
Warning:          10
Inactive:         5
Last Change:      abr 10, 2008
Password Expires: may 20, 2008
Password Inactive: may 25, 2008
Account Expires:  ene 01, 2010
```

 Un usuario cualquiera puede visualizar sus propios detalles, pero se le pedirá su contraseña.

### c. Modificación

Utilice el comando **usermod** para modificar una cuenta. Utiliza la misma sintaxis y las mismas opciones que useradd, pero dispone también de una sintaxis complementaria que necesita algunas precisiones.

| Opción                  | Papel                                                                                                                                                       |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-L</b>               | Bloqueo de la cuenta, como passwd -l.                                                                                                                       |
| <b>-U</b>               | Desbloqueo de la cuenta, como passwd -u.                                                                                                                    |
| <b>-e &lt;n&gt;</b>     | Vencimiento: la contraseña expira n días después del 01/01/1970.                                                                                            |
| <b>-u &lt;UID&gt;</b>   | Modifica el UID asociado al login. Se modifica en consecuencia el propietario de los ficheros que pertenecen al antiguo UID dentro del directorio personal. |
| <b>-l &lt;login&gt;</b> | Modifica el nombre de login.                                                                                                                                |
| <b>-m</b>               | Move: implica la presencia de -d para especificar un nuevo directorio personal. Se mueve el contenido del antiguo directorio al nuevo.                      |

### d. Supresión

Suprime un usuario con el comando **userdel**. Por defecto no se suprime el directorio personal. Para ello, debe pasar la opción **-r**.

```
# userdel -r bean
```

El comando no comprueba la totalidad del sistema de archivos, los archivos que estén fuera de su directorio personal no se borrarán. Es el administrador quien debe buscarlos y borrarlos. Para ello, puede usar el comando **find**.

## 4. Gestión de los grupos

### a. Creación

Puede crear un grupo directamente en el fichero **/etc/group** o bien utilizar los comandos asociados. Si edita el fichero manualmente, utilice el comando **vigr** (o vipw -g).

El comando **groupadd** permite crear un grupo. Su sintaxis sencilla acepta el argumento **-g** para especificar un GID preciso.

```
# grep amigos /etc/group
amigos:!/:1234:
```

## b. Modificación

El comando **groupmod** permite modificar un grupo. Sus parámetros son los siguientes:

| Opción                   | Papel                                                                                              |
|--------------------------|----------------------------------------------------------------------------------------------------|
| <b>-n &lt;nombre&gt;</b> | Renombra el grupo.                                                                                 |
| <b>-g &lt;GID&gt;</b>    | Modifica el GID. Cuidado: no se modifica el grupo al que pertenecen los ficheros correspondientes. |
| <b>-A &lt;user&gt;</b>   | Añade el usuario especificado en el grupo (grupo secundario).                                      |
| <b>-R &lt;user&gt;</b>   | Suprime el usuario especificado del grupo.                                                         |

```
# groupmod -R seb amigos
# grep amigos /etc/group
amigos:!/:1234:
```

## c. Supresión

El comando **groupdel** permite suprimir un grupo. Primero el comando comprueba si el grupo que usted desea suprimir es el grupo primario de un usuario. En este caso, no se permite suprimir el grupo.

Pero si finalmente se puede, no se efectúa más acción que la de suprimir la línea correspondiente en /etc/group. Le corresponde a usted comprobar el sistema de ficheros (y la configuración de las aplicaciones si es necesario) para suprimir cualquier traza de este grupo.

```
# groupdel amigos
```

## 5. Comandos adicionales

### a. Conversión de los ficheros

Seguramente empleará pocas veces los comandos siguientes. Son útiles sobre todo en el contexto de migraciones de servidores Linux hacia otros Unix, y viceversa. Algunos sistemas Unix no utilizan por defecto (hay que activarla después) la gestión de las cuentas con los ficheros **shadow**. En este caso, puede ser necesario convertir los ficheros **/etc/shadow** y **/etc/passwd** en uno único, **/etc/passwd**. Es el papel del comando **pwunconv**.

En el ejemplo siguiente, se ha convertido el fichero **/etc/passwd**. Una vez ejecutado el comando, cualquier traza de **/etc/shadow** ha desaparecido.

```
# pwunconv
# grep bean /etc/passwd
bean:$2a$10$dwbUGrC75bs3l52V5DHxZefkZyB6VTHsLH5ndjsNe/vF/HAzH0cR2:1001:100:
```

```
toto:/home/bean:/bin/bash
# ls -l /etc/shadow
ls: no puede acceder a /etc/shadow: Ningún fichero o directorio de este tipo
```

► Cuidado: el comando **pwunconv** es destructivo. Toda información de la validez de las contraseñas será destruida. Ya no es posible utilizar las diversas opciones de chage o de passwd relativas a los períodos de validez.

El comando **pwconv** hace lo contrario: crea el fichero **/etc/shadow** asociado a **/etc/passwd**, mueve a él las contraseñas y coloca en él los ajustes por defecto, tales como los definidos en el fichero **/etc/login.defs**.

```
# grep bean /etc/passwd
bean:x:1001:100:pepito:/home/bean:/bin/bash
p64p17bicb3:/home/seb # grep bean /etc/shadow
bean:$2a$10$dwbUGrC75bs3l52V5DHxZefkZyB6VTHsLH5ndjsNe/vF/HAzH0cR2:
13984:0:99999:7:::0
```

Los comandos **grpconv** y **grpunconv** hacen lo mismo para los grupos, pero sin éxito, ya que la mayoría de las distribuciones no soportan los grupos en shadow.

## b. Verificar la coherencia

Puede ser útil iniciar herramientas de comprobación de la coherencia de los ficheros de los grupos y de las contraseñas. Si está acostumbrado a modificar estos ficheros manualmente, nada le asegura que todo esté en orden: puede que falte un grupo, que no exista un shell, que un directorio personal esté ausente, etc. El comando **pwck** efectúa una verificación de los ficheros **/etc/passwd** y **/etc/shadow** y reporta los errores.

Veamos un ejemplo en el cual se han alterado los datos del usuario bean de manera voluntaria para probar el comando: el grupo no existe, tampoco el shell. Entre tanto, se ha descubierto otro problema en la máquina de prueba.

```
# pwck
Checking `/etc/passwd'
User `suse-ncc': directory `/var/lib/YaST2/suse-ncc-fakehome' does
not exist.
User `bean': unknown group `14400'
User `bean': shell `/bin/bashr' is not executable.
Checking `/etc/shadow'.
```

comando **grpck** hace lo mismo para los grupos. En este caso, los controles están menos extendidos y se limitan a los duplicados y a la existencia de los usuarios para los grupos secundarios.

```
# grpck
Checking `/etc/group'
Group `users': unknown user `zorg'
```

## c. Comprobar las conexiones

Usted puede trazar las conexiones en su máquina usando dos comandos. El comando **lastlog** se basa en el contenido de **/var/log/lastlog**. Acepta los parámetros **-u** (especificación de un usuario) y **-t** para buscar las conexiones de los últimos n días.

```
$ lastlog -u seb
Username          Port      Latest
seb              pts/4     jue abr 10 15:13:46 +0200 2008
```

comando **last** hace más o menos lo mismo, pero se basa en `/var/log/wtmp`, que provee una información adicional, como el origen de la conexión (IP, nombre de la consola, etc.) y las fechas de conexión y desconexión, así como la duración de conexión y si el usuario sigue conectado.

```
$ last
seb      pts/1                      Tue Apr 15 14:39  still logged in
seb      pts/4                      Tue Apr 15 12:06  still logged in
seb      pts/6                      Tue Apr 15 10:07  still logged in
seb      pts/3                      Mon Apr 14 13:36 - 15:18 (1+01:42)
seb      pts/1                      Thu Apr 10 15:39 - 11:58 (4+20:18)
seb      pts/4      localhost       Thu Apr 10 15:13 - 15:39 (00:25)
seb      pts/4      localhost       Thu Apr 10 15:13 - 15:13 (00:00)
seb      pts/4      localhost       Thu Apr 10 15:12 - 15:13 (00:00)
...
...
```

## d. Acciones de usuario

El usuario dispone de algunas acciones sobre las informaciones de su cuenta. Entre otras cosas, puede:

- cambiar su shell de conexión,
- cambiar sus datos personales,
- cambiar de grupo principal,
- coger la identidad de otra persona.

### Cambiar de shell

El comando **chsh** permite al usuario modificar de manera definitiva (o hasta el próximo comando chsh) el shell de conexión. No puede elegir cualquier cosa. El shell (o cualquier otro comando) debe estar presente en `/etc/shells`. Esta lista está accesible mediante el parámetro `-l` del comando. La modificación se hace dentro de `/etc/passwd`. Sólo root tiene derecho a modificarla para otros usuarios. Se especifica el nuevo shell con el parámetro `-s`.

```
$ chsh -l
/bin/ash
/bin/bash
/bin/bash1
/bin/csh
/bin/false
/bin/ksh
/bin/sh
/bin/tcsh
/bin/true
...
$ id
uid=1004(bean) gid=100(users) ...
$ chsh -s /bin/ksh
Changing login shell for bean.
Contraseña:
Shell changed.
# grep bean /etc/passwd
bean:x:1004:100:pepito:/home/bean:/bin/ksh
```

### Cambiar el comentario

El usuario puede modificar el comentario del fichero `/etc/passwd` usando el comando **chfn**. Es preferible utilizarlo de manera interactiva (se reserva al root el paso de parámetro en modo no interactivo).

```
$ chfn
Changing finger information for bean.
Contraseña:
Enter the new value, or press ENTER for the default
    Full Name: pepito
    Room Number []: Mister Bean
    Work Phone []: 0102030405
    Home Phone []: 0605040302
    Other:
Finger information changed.
$ grep bean /etc/passwd
bean:x:1004:100:Mister Bean,0102030405,0605040302:/home/bean:/bin/bash
```

### **Cambiar de grupo primario**

El comando **newgrp** permite cambiar de grupo primario de manera temporal, con la condición de que el nuevo grupo especificado sea un grupo secundario del usuario o que el usuario disponga de la contraseña del grupo. Utilizado solo, **newgrp** vuelve al grupo de origen. Las modificaciones son temporales, no se modifica el fichero de las contraseñas.

```
$ id
uid=1004(bean) gid=100(users) grupos=16(dialout),33(video),100(users)
$ newgrp video
$ id
uid=1004(bean) gid=33(video) grupos=16(dialout),33(video),100(users)
```

¿Qué ocurre si usted intenta coger como grupo primario a un grupo que no pertenece a sus grupos secundarios y que está protegido por una contraseña? En el ejemplo siguiente, bean intenta coger como grupo primario **grptest**.

```
$ id
uid=1004(bean) gid=100(users) grupos=16(dialout),33(video),100(users)
$ newgrp grptest
Password:
$ id
uid=1004(bean) gid=1000(grptest) grupos=16(dialout),33(video),
100(users),1000(grptest)
```

### **Cambiar de identidad**

El usuario puede adoptar la identidad de otra persona mientras dura un comando o toda una sesión. Se suele tratar de root, ya que, como sabe, nunca debe uno conectarse de manera permanente como root (o al menos hay que evitarlo). Por lo tanto, para las tareas administrativas, hay que convertirse en root (u otro usuario) el tiempo necesario.

El comando **su** (*substitute user*) permite abrir una sesión, o ejecutar un shell, o un comando dado, con otra identidad. Obviamente, usted debe conocer la contraseña de este usuario.

```
su [-c comando] [-s shell] [-] [usuario]
```

Si no se especifica ningún usuario, se utiliza a root.

```
$ id
uid=1000(seb) gid=100(users) ...
$ su
Contraseña:
# id
uid=0(root) gid=0(root) grupos=0(root)
```

Observe que se utiliza por defecto el entorno del usuario de origen. No se carga el entorno de root (o de cualquier otro usuario).

```
# echo $LOGNAME $USER  
seb seb
```

Para cargar el entorno completo del usuario de destino, añada el guión como parámetro:

```
$ su - bean  
Contraseña:  
$ echo $USER $LOGNAME  
bean bean
```

Para ejecutar de manera puntual un comando con otra identidad, utilice **-c**.

```
$ su -c "make install"
```

El comando **sg** (*substitute group*) es idéntico a **su**, pero emplea un nombre de grupo como argumento.

## 6. Configuración avanzada

 Las descripciones de los ficheros siguientes dependen en gran medida de las versiones de los comandos, de sus opciones de compilación y de la política de seguridad aplicada por parte de los editores de las diversas distribuciones. Es posible que algunos ficheros no estén presentes y que otros contengan parámetros diferentes.

### a. /etc/default/useradd

El fichero **/etc/default/useradd** contiene un cierto número de variables que definen las reglas por defecto que se deben aplicar en el momento de crear un usuario:

- su grupo,
- la raíz de su directorio personal (dónde se ubicará éste),
- si está activo o no,
- el shell,
- su grupo o sus grupos secundarios,
- el lugar donde se sitúa el «esqueleto» de las cuentas (estructura básica de un directorio de usuario),
- la creación o no de un spool (bandeja) de correo,
- etc.

```
$ cat /etc.defaults/useradd  
GROUP=100  
HOME=/home  
INACTIVE=-1  
EXPIRE=  
SHELL=/bin/bash  
SKEL=/etc/skel  
GROUPS=video,dialout
```

```
CREATE_MAIL_SP00L=no
```

En vez de editar este comando a mano, puede utilizar los parámetros **--show-defaults** para ver el contenido y **--save-defaults** para modificarlo.

### b. /etc/default/passwd

El fichero **/etc/default/passwd** contiene algunas reglas utilizadas por el comando **passwd** para el cifrado de las contraseñas. Es posible definir reglas de cifrado globales, pero también por tipo de fichero, y pasar algunas opciones según el método.

```
$ cat /etc/default/passwd
# Encriptación por defecto
CRYPT=md5

# Encriptación para los ficheros (/etc/shadow)
CRYPT_FILES=blowfish

# opción para blowfish
BLOWFIS
```

# La impresión

## 1. Principio

Existen tres estándares de impresión en Unix, uno en System V, otro en BSD y un último federador.

Sea cual sea el estándar, el principio de configuración es el mismo. A cada impresora detectada (en general en /etc/printcap) corresponde una cola de espera (**queue**). Un servicio independiente gestiona el conjunto de estas colas de espera. Este mecanismo permite la impresión multiusuario (los trabajos de impresión están en cola de espera, **job queues**) y en red (se puede usar el servicio desde otra máquina remota).

En general, todas las impresoras saber imprimir directamente texto llano ASCII en 80 columnas. Para imprimir documentos formateados o imágenes, se puede utilizar un driver. En realidad se habla de **filtro de impresión**. El filtro puede ser un script o un binario que recupera el flujo entrante (texto, imagen, documento, postscript...), lo identifica y, con la ayuda de tratamientos asociados, lo transforma en lenguaje comprensible por la impresora (PostScript, PCL, Canon, Epson, WPS...).

---

Si tiene la posibilidad y los medios, no dude en elegir una impresora compatible Postscript, que representa una garantía de compatibilidad perfecta. El sitio [linuxprinting.org](http://linuxprinting.org) dispone de una base completa de compatibilidad de las impresoras en Linux.

---

Linux acepta los comandos procedentes de los Unix de tipo System V y BSD. Durante mucho tiempo, el subsistema de impresión se basaba en los servicios BSD y el demonio **lpd**. Desde hace unos años, todas las distribuciones se basan en CUPS, compatible (los comandos, al menos) con los antiguos sistemas de impresión.

## 2. System V

Los comandos de gestión de las colas de espera e impresiones con System V son los siguientes:

- `lp [-dImpresora] [-nCifra] fic1`: imprime el contenido del fichero fic1. La opción `-d` permite elegir la impresora; `-n`, el número de ejemplares.
- `lpstat [-d] [-s] [-t] [-p]`: información relativa a la impresión. La opción `-d` muestra el nombre de la impresora por defecto; `-s`, un resumen de las impresoras; `-t`, la totalidad de la información (estado, trabajos...); `-p [lista]`, únicamente la información sobre las impresoras incluidas en la lista.
- `cancel [ids] [printers] [-a] [-e]`: suprime las tareas de impresión ids de las impresoras printers. La opción `-a` suprime todos los trabajos del usuario; `-e`, todos los trabajos (únicamente para el administrador).
- Se pueden encontrar comandos **enable** y **disable** que emplean como parámetro el nombre de la cola de espera, lo que permite activar o desactivar su acceso.

El demonio suele llevar el nombre de **lpd** (*line printer daemon*) o **Ipsched** (*line printer scheduler*).

- **lpadmin** permite administrar los servicios de impresión como las colas de espera asociadas a una impresora y la cola de espera por defecto. P. ej.: `lpadmin -p cola1 -v impresora-m modelo`.
- `lpadmin -x fila`: supresión de la cola de espera.
- `lpadmin -d fila`: definir la cola de espera por defecto.
- `lpadmin -p fila -u allow:lista`: autorización de imprimir para los usuarios especificados.

- `lpadmin -p fila -u deny:lista`: prohibición de imprimir para los usuarios especificados.
- `lpshut` detiene el servicio de impresión. Cuando se inicia de nuevo el demonio, se reanudan las impresiones en curso en el momento de la parada.
- `accept` y `reject` permiten validar una cola de espera para la impresión o cerrarla.
- `lpmove` permite transferir peticiones de impresión de una cola de espera hacia otra.

### 3. BSD

- `lpr [-dImpresora] [-#copias] fic1`: imprime el contenido del fichero `fic1`. La opción `-P` permite especificar la impresora; `-#`, el número de copias.
- `lpq [-Pimpresora]`: indica el estado y la lista de los trabajos para la impresora especificada si es preciso por la opción `-P`.
- `lprm [-Pimpresora] [-] [ids]`: permite suprimir un trabajo de la impresora especificada por la opción `-P`, la opción `-` suprime todos los trabajos del usuario, `ids` representa una lista de trabajos para suprimir.
- El comando **lpc** es una especie de pequeño shell que permite controlar las impresoras y los trabajos.

El servicio se suele llamar **lpd**.

### 4. CUPS

#### a. Presentación

**CUPS** (*Common Unix Printing System*) es un sistema de impresión de Unix orientado a red:

- Basado en el protocolo **IPP** (*Internet Printing Protocol*), que se basa a su vez en el protocolo **HTTP/1.1**.
- Fácil de utilizar, en particular gracias a una configuración y administración centralizada desde una interfaz HTTP, reglas de conversión basadas en los tipos MIME y ficheros de descripción de impresora estándares (**PPD**, *PostScript Printer Description*).
- CUPS recoge los comandos System V y BSD ya tratados para más sencillez.
- Las trazas de las impresiones están disponibles en el formato **CLF** (*Common Log Format*) de servidores Web y se pueden aprovechar con las mismas herramientas.
- CUPS es capaz de interactuar con los servidores de impresión LPD para guardar una compatibilidad ascendente.
- CUPS dispone de su propia API, que permite crear interfaces de usuario que pueden integrarse en entornos gráficos o interfaces de administración.
- Los pools de impresión permiten redireccionar automáticamente las tareas.
- La autenticación es posible por usuario, máquina o certificado numérico.

El servicio de impresión se llama **cupsd**.

```
$ ps -ef | grep cupsd
```

```
root      6924      1  0 11:20 ?          00:00:00 cupsd -C /etc/cups/cupsd.conf
```

No hacen falta herramientas gráficas para administrar un servidor CUPS. Sin embargo, con el fin de facilitar la administración de la impresión, la mayoría de las distribuciones cuentan con ellas. CUPS propone una interfaz de administración WEB directamente accesible desde el puerto 631 del servidor. La interfaz funciona con cualquiera navegador HTTP.

<http://localhost:631>

El fichero de configuración es `/etc/cups/cupsd.conf`. El equivalente al fichero `/etc/printcap` se encuentra en `/etc/cups/printers.conf`.

```
<DefaultPrinter lj2100>
Info Laserjet 2100
DeviceURI socket://192.168.1.10:9100
State Idle
StateTime 1203806079
Accepting Yes
Shared Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
OpPolicy default
ErrorPolicy stop-printer
</Printer>
```

página de inicio de CUPS no se muestra o el navegador le devuelve un mensaje de error, compruebe en el archivo `/etc/cups/cups.conf` el estado de las líneas siguientes que activan la interfaz web y los permisos de acceso:

```
# Only listen for connections from the local machine.
Listen localhost:631
Listen /var/run/cups/cups.sock

# Show shared printers on the local network.
Browsing On
BrowseOrder allow,deny
BrowseAllow all
BrowseLocalProtocols CUPS dnssd

# Default authentication type, when authentication is required...
DefaultAuthType Basic

# Web interface setting...
WebInterface Yes

# Restrict access to the server...
<Location />
  Order allow,deny
</Location>

# Restrict access to the admin pages...
<Location /admin>
  Order allow,deny
</Location>

# Restrict access to configuration files...
<Location /admin/conf>
  AuthType Default
  Require user @SYSTEM
```

```
Order allow,deny  
</Location>
```

## b. Añadir una impresora

Tiene dos soluciones para añadir una impresora:

- Editar los ficheros a mano.
- Pasar por la interfaz Web o una herramienta de su distribución.

En el primer caso, debe modificar el fichero printers.conf para añadir una sección para su impresora y luego copiar en /etc/cups/PPD el fichero PPD correspondiente a su impresora. Después lo tiene que renombrar con el nombre de sección (p.ej.: lj2100.ppd) del fichero printers.conf. Finalmente, debe buscar la configuración de CUPS. En Red Hat u openSUSE, por ejemplo:

```
# service cups reload
```

Si la interfaz aparece en inglés, compruebe en el archivo de configuración que el parámetro DefaultLanguage alberga el valor es. También tiene que comprobar el lenguaje por defecto del navegador. Si a pesar de todo, el idioma de la interfaz sigue siendo el inglés, entonces compruebe si hay disponible un paquete de idiomas adicional para su distribución. En el peor de los casos, la interfaz sigue siendo comprensible (trabajos = jobs, impresoras = printers, etc.).

Como se suele activar la interfaz Web por defecto, puede pasar por un navegador Web. Es posible que durante el acceso a las páginas de administración se le requiera identificación. En principio, los de root son suficientes, pero puede crear cuentas (o añadir usuarios) encargados de la gestión de las impresiones con el comando **Ippasswd**.

```
$ sudo lppasswd -a seb  
Introduzca la contraseña:  
Introduzca nuevamente la contraseña:
```

En la página principal, haga clic en la pestaña **Administración**.

C UPS y el logo de CUPS son marcas registradas de Apple, Inc. Los derechos de copia de CUPS 2007-2011 son de Apple Inc. Todos los derechos reservados.

## *Interfaz de administración de CUPS*

Es posible que CUPS detecte las impresoras locales o en red cuando éstas cuentan con servicios LPD (puerto 9100) o IPP, ya que el servicio está siempre a la escucha. En el ejemplo actual, se ha detectado una impresora y el hecho de hacer clic en **Añadir esta impresora** facilita ampliamente la tarea.

En este ejemplo, basado en añadir manualmente una impresora HP Laserjet 2100, añadirá una impresora al hacer clic en **Añadir impresora**.

- La etapa siguiente consiste en elegir su tipo de conexión. La impresora está conectada en red. En este caso seleccione **AppSocket/HP JetDirect**. Haga clic en **Siguiente**.



## *Conexión de una impresora de red*

Para una impresora en red, tiene varias opciones. La mayoría de las impresoras conectadas a una red de tipo Ethernet o WiFi ofrecen servicios de impresión LPD o Socket (impresión directa). En este último caso, es la propia impresora la que gestiona los trabajos de impresión. Para el resto (IPP, http, Samba, etc.), deberá consultar el manual de su impresora. Algunas impresoras profesionales ofrecen una interfaz de configuración Web (como los routers que tenemos en casa) para configurar los puertos.

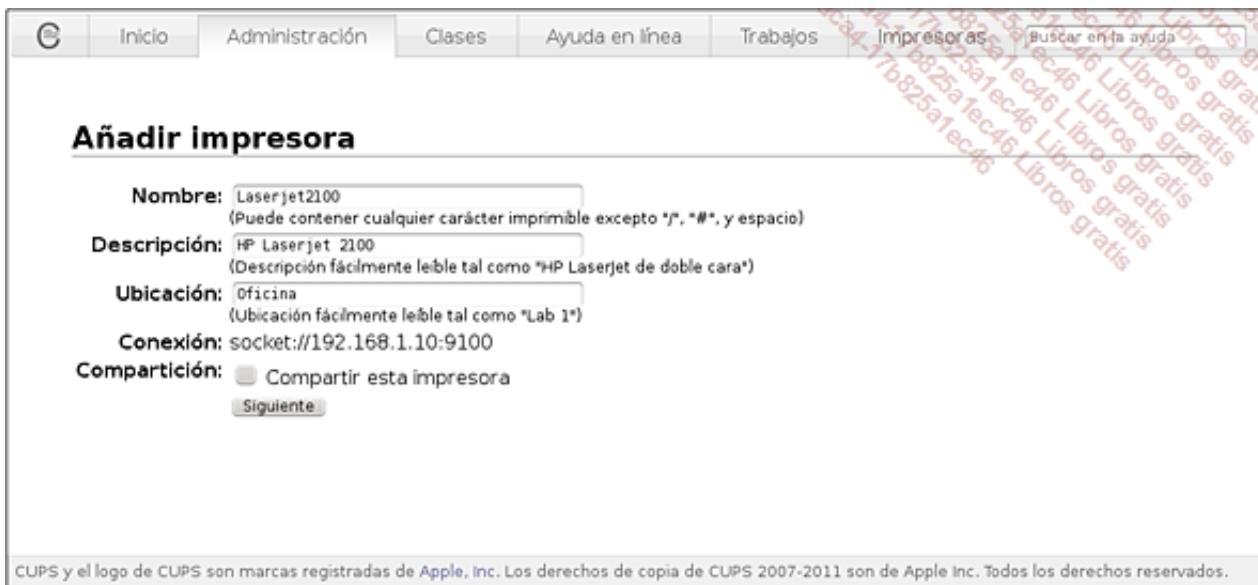
Si su impresora remota ya está configurada en otro servidor CUPS, puede pasar este servidor y las **URI** de tipo http, ipp o lpd para imprimir por ellos.

La impresora que usaremos en este ejemplo es la Laserjet 2100 que dispone de una tarjeta de red y de un servidor de impresión jetDirect integrados. La URI introducida es `socket://192.168.1.10:9100`.

La IP es la de la impresora de red, el puerto 9100 es el puerto estándar en este caso.

A continuación haga clic en **Siguiente**.

- Inserte la información siguiente:
    - El **nombre** de la impresora es el nombre de la espera que será visible en las herramientas de impresión. No ponga espacio.
    - Se puede dejar la **ubicación** vacía, pero puede ser útil para localizar la impresora (en el caso de una impresora remota).
    - La **descripción** puede ser lo que usted desee, como por ejemplo el modelo completo de la impresora.



CUPS y el logo de CUPS son marcas registradas de Apple, Inc. Los derechos de copia de CUPS 2007-2011 son de Apple Inc. Todos los derechos reservados.

*Nombre de la impresora*

Haga clic en **Siguiente**.

- Ahora elija un driver que corresponde al modelo de su impresora. Observará que para varios modelos existen varios drivers. Se suele indicar el driver aconsejado con **recommended**; tendrá que comprobar el driver realmente recomendado para su impresora en el sitio siguiente:

<http://www.linux-foundation.org/en/OpenPrinting>

Que informa de que, para esta impresora:

For basic printing functionality use the PostScript PPD. For advanced functionality such as printer status and maintenance features, use the **HPLIP** driver (which includes **HPIJS**).

Por lo tanto, se debe elegir el driver **HP Laserjet 2100 hpijs**.

---

 Linux soporta muy bien las impresoras (y los escáneres) de la marca HP a través del proyecto libre del editor **HPLIP** (*HP Linux Imaging and Printing*), incluyendo las soluciones integradas (impresora, escáner, fax, fotocopiadora, impresión fotográfica, etc.). La lista del hardware soportado está accesible mediante <http://hplip.sourceforge.net/>. Por supuesto, otros fabricantes ofrecen impresoras perfectamente compatibles con Linux, como Samsung.

---

Es posible especificar otro fichero PPD. Varios fabricantes proponen este tipo de ficheros para su impresora. Se presentan los ficheros PPD por defecto en /usr/share/cups/model.

- Haga clic en **Añadir impresora**.

Inicio Administración Clases Ayuda en línea Trabajos Impresoras Buscar en la ayuda

## Añadir impresora

**Nombre:** Laserjet2100  
**Descripción:** HP Laserjet 2100  
**Ubicación:** Oficina  
**Conexión:** socket://192.168.1.10:9100  
**Compartición:** No compartir esta impresora  
**Marca:** HP   
**Modelo:**

**O proporcione un archivo PPD:**

CUPS y el logo de CUPS son marcas registradas de Apple, Inc. Los derechos de copia de CUPS 2007-2011 son de Apple Inc. Todos los derechos reservados.

### Selección del driver de impresión

- Llega a la pantalla que presenta la lista de las impresoras actualmente configuradas. Puede probar su impresora imprimiendo la página de prueba. También puede modificar las opciones por defecto de la impresora. Estas opciones son de varios tipos:
  - Opciones generales para indicar el tipo de papel por defecto (A4), la calidad de salida, la bandeja por defecto (las Laserjet 2100TN tienen dos bandejas y una entrada frontal), la impresión en Duplex (si está el kit), etc.
  - El modo de salida y las diferentes opciones, para ajustar por ejemplo la resolución y el tipo de papel por defecto, un filtro cualquiera, etc.
  - Las marcas de separación: para separar las hojas (en el caso de que se use papel continuo), usted puede ubicar la marca antes y después de la impresión. Se puede personalizar el fichero de marcas de impresión hasta cierto punto (vea el manual de CUPS para ello).
  - La facturación. Se puede configurar CUPS para gestionar la facturación por servicio/máquinas/usuarios, en el caso de que se utilicen algunos servicios avanzados.

Establisher opciones predeterminadas de Laserjet2100

Consultar a la impresora las opciones predeterminadas

General    Opciones instaladas    Rótulos    Reglas

**General**

Media Size: A4  
Media Source: Default  
Output Mode: Color  
Output Resolution: 600 DPI  
Double-Sided Printing: Off

Cambiar opciones predeterminadas

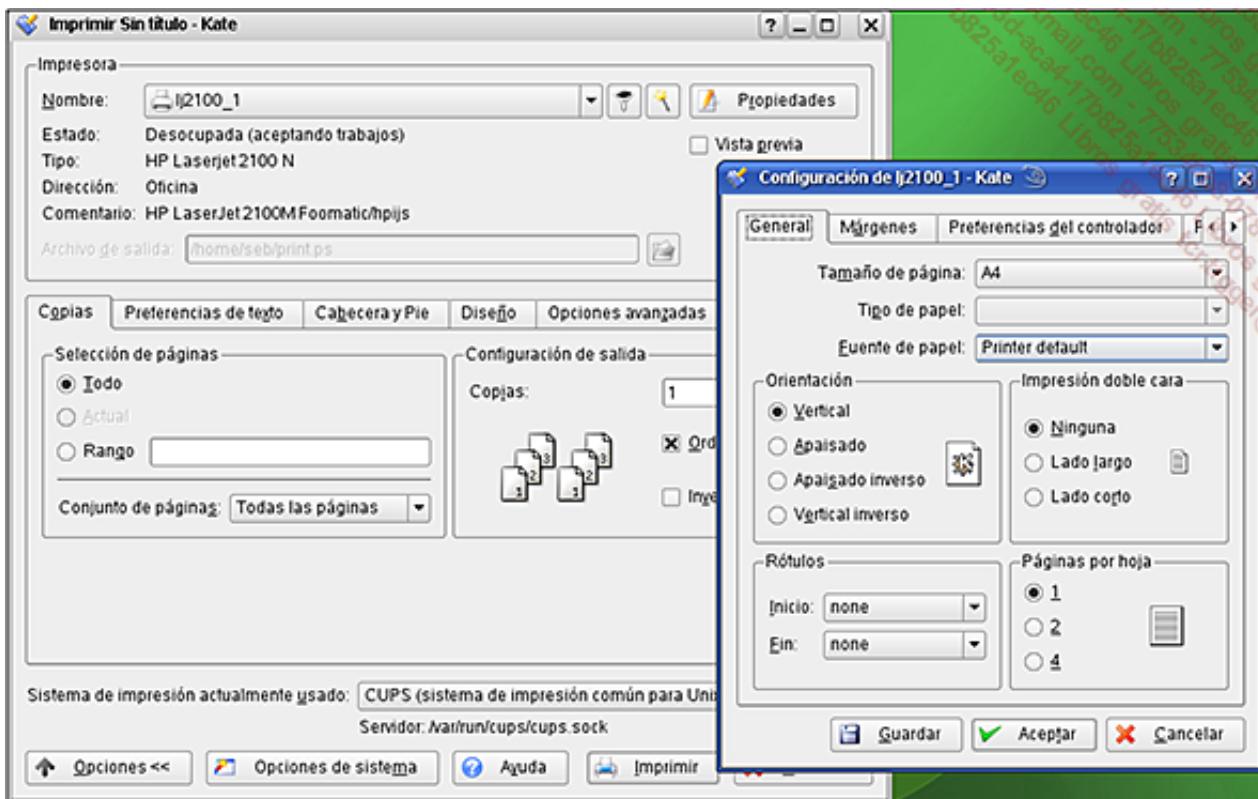
CUPS y el logo de CUPS son marcas registradas de Apple, Inc. Los derechos de copia de CUPS 2007-2011 son de Apple Inc. Todos los derechos reservados.

### *Configuración de la impresora*

Tendría que dejar, además del formato de papel, las opciones por defecto. La mayoría de los entornos de escritorio y programas proponen modificar las opciones de impresión mediante CUPS, como en los sistemas de tipo Windows.

La captura siguiente fue efectuada en KDE. En **Konqueror**, se llamó a la entrada **Imprimir** del menú **Fichero** durante la visualización de una imagen. Todas las opciones accesibles por el botón **Propiedades** al lado del nombre de la impresora proceden de las posibilidades propuestas por CUPS y el driver de impresión PPD. Dos impresoras distintas, e incluso dos drivers distintos (si varios están disponibles), no tendrán las mismas opciones.

Si hace clic en la pestaña **Configuración del driver**, obtendrá las mismas opciones que las propuestas por la interfaz Web de CUPS.



*La impresión en KDE*

# Automatización

## 1. Con cron

### a. Presentación

El servicio **cron** permite la programación de eventos repetidos. Funciona con la ayuda de una tabla, llamada una **crontab**. Es un fichero de texto que se puede editar con un simple editor, como por ejemplo vi. Para modificar su crontab personal, utilice el comando **crontab** con objeto de editar la tabla con el parámetro -e.

Se guardan los ficheros contrabs en /var/spool/cron.

El servicio **cron** debe estar en ejecución para que las crontabs estén activas.

```
$ ps -ef|grep cron
root      3634      1  0 18:28 ?
          00:00:00 /usr/sbin/cron
```

### b. Formatos

El formato de un registro de crontab es el siguiente:

Minutos	Horas	Día del mes	Mes	Día semana	Comando
1	2	3	4	5	6

Utilice el formato siguiente para los valores periódicos:

- Un valor para indicar cuándo se debe ejecutar el comando. P. ej.: el valor 15 en el campo minuto significa el decimoquinto minuto.
- Una lista de valores separados por comas. P. ej.: 1,4,7,10 en el campo mes para enero, abril, julio, octubre.
- Un intervalo de valores. P. ej.: 1-5 en el campo día de la semana indica de lunes (1) a viernes (5). El 0 es el domingo y el 6, el sábado.
- El carácter \* para todos los valores posibles. P. ej.: \* en el campo día del mes indica todos los días del mes o de los meses.

### c. Ejemplos

Ejecución de df todos los días, todo el año, cada cuarto de hora:

```
0,15,30,45 * * * * df > /tmp/libre
```

Ejecución de un comando todos los días laborables a las 17 horas:

```
0 17 * * 1-5 fin_trabajo.sh
```

Listar las crontabs activas:

```
$ crontab -l
```

#### Suprimir la crontab activa:

```
$ crontab -r
```

#### Editar la crontab de un usuario particular:

```
# crontab -u user
```

### d. crontab sistema

La configuración crontab general para el sistema se encuentra en `/etc/crontab`. Su sintaxis difiere un poco. Hay un campo adicional. Permite especificar el usuario con el que se ejecutará el comando.

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

Aquí todos los días a las 4:02 h de la mañana se ejecuta `run-parts /etc/cron.daily`. El script **run-parts** acepta como parámetro un directorio y ejecuta todos los programas presentes en éste.

```
$ ls cron.daily/
00-logwatch 0anacron  makewhatis.cron  slocate.cron
00webalizer logrotate  rpm  tmpwatch
```

Entre los programas ejecutados, fíjese **logrotate**, que permite efectuar copias de seguridad y renombrar ficheros logs y ficheros de diario del sistema con el fin de que éstos no se vuelvan inabarcables a consecuencia de su tamaño. El programa **tmpwatch** se encarga de limpiar el sistema de los ficheros «ociosos» (en `/tmp` por ejemplo).

Para terminar, el directorio `/etc/cron.d` contiene crontabs adicionales siempre en formato crontab sistema. Por lo tanto, una aplicación que debe ejecutar tareas recurrentes deberá crearse su propio archivo en vez de modificar una tabla existente.

### e. Control de acceso

Usted puede controlar el acceso al comando **crontab** por usuario con los ficheros `/etc/cron.allow` y `/etc/cron.deny`.

- Si `cron.allow` está creado, sólo los usuarios explícitamente indicados pueden utilizar **cron** (ver Automatización - Con at, en este capítulo).
- Si `cron.allow` no está creado, `crontab` comprueba la presencia de un fichero `cron.deny`. Todos los usuarios que no estén en él están autorizados a utilizar `crontab`. Si está vacío, todo el mundo está autorizado a utilizar **cron**.
- Si no existe ninguno de los dos ficheros, sólo `root` puede utilizar `crontab`.

## 2. Con at

## a. Presentación

El comando **at** y los comandos asociados permiten la gestión de los batchs. A diferencia de la contrab, los modificaciones son volátiles: se pierden cuando se termina la sesión. Le corresponde a usted colocar la lista de los comandos en un posible fichero y cargarlo si es necesario mediante los scripts de su perfil.

Para que funcione at, el servicio **atd** (*at daemon*) debe estar en marcha.

```
$ ps -ef | grep atd  
at      7988      1  0 21:05 ?          00:00:00 /usr/sbin/atd
```

## b. Formatos

Para simplificar, hay dos maneras de utilizar at:

- pasándole una línea de comandos de manera interactiva,
- pasándole un fichero ejecutable que contiene los comandos que se deben ejecutar.

En los dos casos, debe facilitar a at una hora de ejecución. El formato de esta hora es bastante flexible. Para programar la ejecución de una línea de comandos a las 21:20 h de manera interactiva:

```
$ at 21:20  
warning: commands will be executed using /bin/sh  
at> echo hola  
at> <EOT>  
job 4 at 2008-05-08 21:20
```

Después de haber entrado el comando o los comandos que se deben ejecutar a las 21:20 h, pulse [Intro], y en una línea vacía pulse [Ctrl] D (final de entrada). El comando **at** confirma la programación del comando.

Para programar la ejecución de un comando (script o binario) a las 21:25 h:

```
$ at -f /home/seb/test.sh 21:25  
warning: commands will be executed using /bin/sh  
job 6 at 2008-05-08 21:25
```

### Hora

Se puede formatear la hora de la manera siguiente:

- HHMM o HH:MM.
- La hora puede tener el formato de 12 o 24 h. Con el formato de 12 horas, puede especificar AM (mañana) o PM (tarde).
- Midnight (medianoche), noon (mediodía), teatime (16:00 h, típicamente inglés).
- MMJJAA, MM/JJ/AA o JJ.MM.AA para una fecha particular.
- Now: ahora.
- + n minutes/hours/days/weeks: la hora actual a la que se añaden n minutos/horas/días/semanas.

Si la hora especificada es inferior a la hora actual, se ejecuta el comando al día siguiente.

```
$ at 21:30 09.05.2008  
warning: commands will be executed using /bin/sh  
at> echo ihola!
```

```
at> <EOT>
job 9 at 2008-05-09 21:30
$ at now + 2 days
warning: commands will be executed using /bin/sh
at> echo en dos días
at> <EOT>
job 10 at 2008-05-10 21:29
```

- También existe el comando **batch**, que no tiene en cuenta la hora. Ejecuta el comando en cuanto el nivel de trabajo de la máquina lo permite. Se puede especificar la hora, pero en este caso se considera como «a partir de esta hora en cuanto sea posible».

### c. Control de las tareas

El comando **atq** (*at queue*) permite listar las tareas programadas:

```
$ atq
10      2008-05-10 21:29 a seb
9       2008-05-09 21:30 a seb
```

Se colocan los jobs (tareas) en el directorio */var/spool/atjobs*, a razón de un ejecutable por tarea.

```
# ls -l /var/spool/atjobs/
-rwx----- 1 seb users 5620 may  8 21:29 a000090133cf92
-rwx----- 1 seb users 5628 may  8 21:30 a0000a0133d531
```

Si mira el contenido del ejecutable, verá que su comando no está solo. Está situado al final, pero el script ubica todo el entorno durante la creación de la entrada at.

```
#cat a0000a0133d531
#!/bin/sh
# atrun uid=1000 gid=100
# mail      seb 0
umask 22
LESSKEY=/etc/lesskey.bin; export LESSKEY
NNTPSERVER=news; export NNTPSERVER
INFODIR=/usr/local/info:/usr/share/info:/usr/info; export INFODIR
MANPATH=/usr/local/man:/usr/share/man:/opt/gnome/share/man; export
MANPATH
KDE_MULTIHEAD=false; export KDE_MULTIHEAD
... (unas 80 líneas) ...
cd /home/seb || {
    echo 'Execution directory inaccessible' >&2
    exit 1
}
echo ihola!
```

comando **atrm** permite suprimir una tarea:

```
$ atrm 10
$ atrm 9
$ atq
```

### d. Control de acceso

Es posible controlar el acceso al comando **at** por usuario con los ficheros */etc/at.allow* y */etc/at.deny*.

- Si at.allow está creado, sólo los usuarios explícitamente indicados pueden utilizar at.
- Si at.allow no está creado, at comprueba la presencia de un fichero at.deny. Todos los usuarios que no están en él están autorizados a utilizar at. Si está vacío, todo el mundo está autorizado a utilizar el comando at.
- Si no existe ninguno de los dos ficheros, sólo root puede utilizar at.

# Los registros (logs) del sistema

## 1. Fundamentos

Cuando el sistema se inicia, se pone en marcha y efectúa cualquier tipo de acción, se registran sus acciones y las de la mayoría de sus servicios en diferentes ficheros. Dos servicios están especializados en la recepción de los mensajes que tienen como destino estos ficheros:

- **klogd**: *kernel log daemon*, encargado de la gestión de la información emitida por el núcleo.
- **syslogd**: *system log daemon*, encargado de la gestión de la información emitida por cualquier tipo de servicio y, si procede, del núcleo.

 Ciertas distribuciones utilizan ahora **syslog-ng** o **rsyslog**, cuyas reglas de tratamiento de los mensajes, basadas en expresiones regulares, han evolucionado considerablemente. Sin embargo, el principio sigue siendo exactamente el mismo.

Desde un punto de vista histórico, el servicio syslogd gestionaba también los mensajes emitidos por el núcleo. Todavía es capaz de hacerlo, pero la cantidad de mensajes emitidos, los diferentes niveles de gravedad y los nuevos métodos de acceso a los mensajes del núcleo hacen que parezca importante y pertinente separar la gestión de los mensajes del núcleo, y la de los emitidos por los servicios.

Los mensajes importantes emitidos por un componente del sistema deberían pasar por el servicio syslogd. Esto no impide, al contrario, que un servicio pueda gestionar sus propias entradas en sus propios ficheros. No se deberían colocar entradas de aplicaciones en el registro del sistema. Las trazas de acceso a las páginas Web de un servidor Apache no tienen nada que hacer ahí. En cambio, las de conexión al sistema (mediante la consola, ssh, telnet, etc.) tienen mucho interés y deben estar presentes en los ficheros logs del sistema.

A partir de este punto y en el resto del libro, las entradas en el registro se llamarán por su nombre de uso corriente: los **logs**.

## 2. Los mensajes

El servicio **klogd** gestiona los mensajes emitidos por el núcleo. Dispone de dos fuentes de acceso a los mensajes:

- el sistema de ficheros virtual **/proc**, utilizado por defecto si está presente, y en particular `/proc/kmsg`;
- las llamadas al sistema mediante la API del núcleo, en particular `sys_syslog`, si **/proc** está ausente o si se ha pasado el parámetro `-s` a klogd.

Los mensajes del núcleo tienen niveles de prioridad diferentes, escalados de 0 (alta prioridad) a 7 (mensaje de depuración):

Nivel	Alias sistema	Significado
<b>0</b>	<b>EMERG</b>	No se puede utilizar el sistema.
<b>1</b>	<b>ALERT</b>	Se debe efectuar una acción inmediatamente.
<b>2</b>	<b>CRIT</b>	Problema crítico.

<b>3</b>	<b>ERR</b>	Error.
<b>4</b>	<b>WARNING</b>	Aviso.
<b>5</b>	<b>NOTICE</b>	Normal, pero necesita una atención particular.
<b>6</b>	<b>INFO</b>	Información estándar.
<b>7</b>	<b>DEBUG</b>	Traza de depuración del núcleo.

El servicio **klogd** devuelve los mensajes de nivel 0 a 6 a **syslogd**, que los redirigirá a los ficheros de logs y si es preciso a las consolas correspondientes. Por defecto, no se traza la información de depuración de nivel 7.

El servicio **syslogd** (o syslog-ng) recibe los mensajes procedentes de los servicios y también de klogd. Luego los reparte según el transmisor, la gravedad, en ficheros, en consolas, en forma de mails a los usuarios del sistema (root, por ejemplo), etc.

Las acciones más corrientes son la escritura de los logs en ficheros, la redirección de mensajes hacia una consola (muchas veces se trata de la 10 o 12) o el envío de mensajes a root.

### 3. Configuración de syslog

El fichero de configuración `/etc/syslog.conf` permite definir el origen, la importancia y el destino de cada mensaje, en forma de dos campos.

- El origen define de hecho un conjunto de **sistemas** y de **subsistemas** (núcleo, servicios). La lista, extensible, se compone en el origen de los elementos siguientes. El asterisco define el conjunto de los subsistemas.

<b>Subsistemas</b>	<b>Significado</b>
<b>auth/authpriv</b>	Servicio de seguridad y autenticación.
<b>cron</b>	Servicio cron.
<b>daemon</b>	Los demonios del sistema.
<b>kern</b>	El núcleo.
<b>lpr</b>	El servicio de impresión.
<b>mail</b>	La mensajería.
<b>news</b>	La red.
<b>syslog</b>	El propio Syslog.
<b>user</b>	Mensajes de los procesos de los usuarios.
<b>uucp</b>	Unix to Unix CoPy.



Usted mismo puede mandar, directamente o en sus scripts, mensajes a **syslogd** mediante el comando **logger**.

## 4. El caso de rsyslog

El gestor de registros **rsyslog** se está usando cada vez más como sustituto de syslog y syslog-ng. Su configuración principal se encuentra en **/etc/rsyslog.conf**. Las configuraciones adicionales se encuentran en **/etc/rsyslog.early.conf**, utilizado antes de la activación de las capas de red, y el contenido de **/etc/rsyslog.d/\*** se carga con el archivo de configuración principal.

Además de la sintaxis clásica de syslog, rsyslog dispone de módulos y de una sintaxis más completa. Esta sintaxis permite especialmente redirigir los mensajes en función de reglas más complejas que pueden describirse con la ayuda de comandos de programación. La redirección de los mensajes no se hace más fácil sólo en función de los niveles y subsistemas, sino que también analiza el contenido de los mensajes. La siguiente configuración tiene como objetivo redirigir los mensajes provenientes de la aplicación Network Manager a su propio archivo de registro. Lo hará a partir del nombre o de la porción de nombre del programa emisor.

```
if      ($programname == 'NetworkManager') or \
        ($programname startsWith 'nm-') \
then    -/var/log/NetworkManager
&      ~
```

Los archivos de syslog son compatibles con rsyslog, pero no hay reciprocidad: syslog no puede leer la sintaxis extendida de rsyslog. Se puede iniciar rsyslog en modo compatibilidad con el parámetro **-c0**.

## 5. Los registros

Por convención se ubican los logs del sistema en **/var/log**. No todos los logs de este directorio provienen de **syslogd**. Es el caso, por ejemplo, de la información de conexión. Veamos un ejemplo del contenido de este directorio. Contiene varios ficheros de texto y directorios. Unos servicios pueden decidir, sin pasar por **rsyslogd**, concentrar y escribir sus mensajes en esta estructura.

```
# cd /var/log ; ls -l
-rw-r---- 1 root  root  2460 feb  7 05:34 acpid
drwxr-x--- 2 root  root  4096 mar  5 2007 audit
-rw----- 1 root  root   116 mar 27 04:02 boot.log
-rw----- 1 root  root 75487 mar 28 11:10 cron
drwxr-xr-x 2 lp    sys   4096 mar 27 04:02 cups
-rw-r--r-- 1 root  root 28359 feb  7 05:34 dmesg
drwx----- 2 root  root  4096 ago  7 2007 httpd
-r----- 1 root  root 18747276 mar 28 11:08 lastlog
drwxr-xr-x 2 root  root  4096 jun  1 2007 mail
-rw----- 1 root  root  4537 mar 28 04:02 maillog
-rw----- 1 root  root 178348 mar 28 11:10 messages
drwx----- 2 root  root  4096 oct 16 23:21 samba
-rw----- 1 root  root 214999 mar 28 11:08 secure
-rw-r--r-- 1 root  root  2734 mar 28 11:01 snmpd.log
-rw----- 1 root  root     0 mar 23 04:02 spooler
drwxr-x--- 2 squid  squid  4096 ene 22 2007 squid
-rw----- 1 root  root  62165 mar 28 09:13 sudo.log
drwxr-xr-x 2 root  root  4096 oct  5 2004 vbox
-rw-rw-r-- 1 root  utmp 127872 mar 28 11:10 wtmp
-rw----- 1 root  root  40557 mar 28 11:03 xferlog
```

# Archivado y backup

## 1. Las herramientas de copia de seguridad

La copia de seguridad constituye un trabajo importante del administrador, ya que, en caso de problema grave, se suele restaurar el sistema desde una copia de seguridad, o una imagen del sistema cuando éste estaba todavía bien (funcionamiento correcto, sin corrupción). Se entrega cada Unix con sus propios comandos y procedimientos de copia de seguridad; sin embargo, existen algunas herramientas comunes.

### a. Comandos, planes, scripts

- Para la copia de seguridad de los ficheros y estructuras, utilice los comandos **tar** y **cpio**. Estos comandos guardan una estructura, y no un sistema de ficheros. Se puede hacer coincidir los dos.
- Para la copia de seguridad física de discos y sistemas de ficheros (los dumps), utilice el comando **dd**.

Una copia de seguridad incremental consiste en salvaguardar una primera vez la totalidad de los datos y luego únicamente los ficheros modificados. También se pueden encontrar en forma de programas libres o en el mercado soluciones más avanzadas de copia de seguridad (Networker, por ejemplo).

A veces el administrador tendrá que definir scripts de copia de seguridad y restauración adaptados a cada caso concreto (partición del sistema, datos aplicativos...) y automatizar cuando sea posible la ejecución de estos scripts en función de la fecha, hora o carga de la máquina.

También será importante definir un plan de copia de seguridad y, para ello, hay que hacerse las preguntas correctas:

- ¿Qué se debe salvaguardar?
- ¿Con qué frecuencia?
- ¿Cuánto tiempo se conservarán las copias de seguridad, dónde, cuántos ejemplares?
- ¿Dónde se almacenará el historial de las copias de seguridad?
- ¿Cuál es el soporte más apropiado?
- ¿Cuánta capacidad necesita el soporte de copia de seguridad?
- ¿Cuánto tiempo se prevé para salvaguardar un fichero, un sistema de ficheros?, y ¿es razonable?
- ¿La copia de seguridad debe ser automática o manual?
- ¿Cuál es el método más apropiado para la copia de seguridad?

Como cada caso es único, este libro no puede contestar a todas estas preguntas. Las respuestas dependen del entorno de destino (producción, integración, tests, etc.). Sin embargo, considere siempre en la implementación de una copia de seguridad del sistema (raíz, /opt, /usr, /var, /boot, etc.) después de una instalación y antes de una modificación importante, por si acaso fuera necesario volver atrás.

### b. Otros comandos

- **mt**: control de una cinta magnética.
- **touch**: pone la fecha de última modificación a la hora actual, para forzar una copia de seguridad incremental.

- **find**: selecciona los ficheros que hay que salvaguardar.
- **compress** y **uncompress**: compresión y descompresión de los ficheros.
- **gzip, gunzip, zcat**, compresión y descompresión en formato GnuZip.

## 2. tar

El comando **tar** es simple y eficaz. Crea archivos de los ficheros, incluida la estructura de ficheros, en cualquier tipo de soporte, incluido en otro fichero (archivo con la extensión .tar). El archivo así creado puede extenderse a varios volúmenes: cuando la cinta o el disquete está lleno, le corresponde al usuario insertar uno nuevo. El proceso de copia/restauración continúa.

### a. Archivadores

La sintaxis es la siguiente:

```
tar cvf nombre_archivo Fichero(s)
```

Por ejemplo, para colocar en un archivo **tar** el directorio Desktop:

```
$ tar cvf desktop.tar Desktop/
Desktop/
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
```

Los parámetros son los siguientes:

- **c**: creación de archivo,
- **v**: modo «verboso», **tar** muestra lo que hace,
- **f**: el parámetro siguiente es el nombre del archivo.

### b. Listar

La sintaxis es:

```
tar tvf nombre_archivo
```

Para listar el contenido del archivo anterior:

```
$ tar tvf desktop.tar
drwx----- seb/users          0 2008-04-17 09:44 Desktop/
-rw-r--r-- seb/users      191 2007-10-20 20:10 Desktop/fusion-icon.desktop
-rw-r--r-- seb/users     4786 2007-09-26 00:43 Desktop/konsole.desktop
-rw-r--r-- seb/users      665 2008-04-08 15:14 Desktop/Support.desktop
```

```

-rw-r--r-- seb/users    1051 2007-10-05 10:16 Desktop/Office.desktop
-rw-r--r-- seb/users    4586 2007-12-05 11:37 Desktop/Terminal.desktop
-rw-r--r-- seb/users     829 2007-10-17 12:12 Desktop/MozillaFirefox.desktop
-rw-r--r-- seb/users    3952 2007-10-05 10:16 Desktop/Printer.desktop
-rw-r--r-- seb/users    2053 2007-10-05 10:16 Desktop/.directory
-rw-r--r-- seb/users     450 2007-10-23 11:58 Desktop/myComputer.desktop
-rw-r--r-- seb/users    218 2008-02-22 08:43 Desktop/trash.desktop
-rw-r--r-- seb/users    328 2008-04-08 15:14 Desktop/SuSE.desktop
-rw-r--r-- seb/users    472 2008-04-17 09:44 Desktop/Windows.desktop

```

parámetro t lista el contenido del archivo.

### c. Restauración

Para restaurar el contenido de un archivo, la sintaxis es:

```
tar xvf nombre_archivo ficheros
```

Para restaurar el archivo precedente:

```

tar xvf desktop.tar
Desktop/
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop

```

parámetro x permite la extracción del conjunto de los ficheros del archivo, o del fichero o de los ficheros especificados después del nombre del archivo.

### d. Otros parámetros

El comando **tar** de gnu permite gestionar los formatos de compresión directamente:

- z: se comprime el archivo al formato gzip.
- Z: se comprime el archivo al formato compress.
- j: se comprime el archivo al formato bzip2.

Así, los comandos anteriores para el formato de compresión **gzip** se convierten en:

```

$ tar cvzf desktop.tar.gz Desktop/
Desktop/
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop

```

```
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
$ ls -l desktop.tar*
-rw-r--r-- 1 seb users 30720 may  9 11:16 desktop.tar
-rw-r--r-- 1 seb users  7556 may  9 11:22 desktop.tar.gz
```

Observe la diferencia de tamaño. Se pueden utilizar las opciones de compresión con `c`, `t` y `x`. Observe que se comprime el archivo final, y no los ficheros de manera individual. Puede ser preferible no especificar una opción de compresión si está haciendo una copia en un soporte que gestiona por sí mismo la compresión.

Si comprime su archivo para destinarlo a otros sistemas, o desea guardar una compatibilidad con los parámetros por defecto de tar, puede proceder como a continuación:

```
$ gzip -cd desktop.tar.gz | tar xvf -
Desktop/
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
```

parámetro `-d` precisa a **gzip** que debe descomprimir el fichero, mientras que `-c` pasa el resultado por la salida estándar. El `-` final indica a tar que debe recuperar el flujo por la entrada estándar.

### 3. cpio

El comando **cpio** hace una copia de seguridad sobre la salida estándar de los ficheros cuyos nombres están insertados en la entrada estándar, por defecto la pantalla y el teclado. Por lo tanto, es posible utilizar las redirecciones. **cpio** no comprime los archivos. Le corresponde a usted hacerlo.

#### a. Archivar

La sintaxis general es:

```
cpio -oL
```

Los parámetros más utilizados son:

- `-o`: output, creación de la copia de seguridad en salida.
- `-L`: copia los ficheros relacionados, y no los vínculos simbólicos.
- `-v`: modo «verboso», información detallada.
- `-c`: copia atributos de los ficheros en forma ASCII (para el intercambio entre varios SO).

Veamos cómo archivar y comprimir el directorio Desktop:

```
find Desktop -print| cpio -ocv | gzip > archive.cpio.gz
Desktop
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
42 blocks
> ls -l archive.cpio.gz
-rw-r--r-- 1 seb users 7377 may 9 11:33 archive.cpio.gz
```

## b. Listar

La sintaxis general es:

```
cpio -it archivo
```

Las parámetros son:

- **-i**: lectura del archivo en entrada.
- **-t**: como para tar, lista el contenido del archivo.

```
$ cat archive.cpio.gz | gzip -cd | cpio -it
Desktop
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
42 blocks
```

## c. Restaurar

La sintaxis general es:

```
cpio -i[umd]
```

- **-u**: restauración incondicional, con borrado de los ficheros que ya existen. Por defecto no se restauran los ficheros si los que están presentes en el disco tienen como fecha de última modificación la misma o posterior.
- **-m**: los ficheros restaurados conservan su última fecha de modificación.
- **-d**: cpio reconstruye el árbol de directorios y subdirectorios que faltan.

Para restaurar el archivo precedente:

```
$ cat archive.cpio.gz | gzip -cd | cpio -iuvd
Desktop
Desktop/fusion-icon.desktop
Desktop/konsole.desktop
Desktop/Support.desktop
Desktop/Office.desktop
Desktop/Terminal.desktop
Desktop/MozillaFirefox.desktop
Desktop/Printer.desktop
Desktop/.directory
Desktop/myComputer.desktop
Desktop/trash.desktop
Desktop/SuSE.desktop
Desktop/Windows.desktop
42 blocks
```

## 4. dd

El comando **dd** (*device to device*) tiene como misión la copia física, bloque por bloque, de un fichero periférico hacia un fichero periférico. En el origen se utilizaba para la lectura y escritura en cinta magnética, pero se puede utilizar también con cualquier fichero. El comando **dd** permite realizar copias físicas de discos y sistemas de ficheros.

Argumento	Papel
<b>if=fichero</b>	Nombre del fichero en entrada (el que se debe copiar)
<b>of=fichero</b>	Nombre del fichero en salida.
<b>bs=n</b>	Tamaño del bloque en bytes.
<b>count=n</b>	Número de bloques para copiar.
<b>skip=n</b>	Número de bloques que hay que saltar al principio del fichero de entrada.
<b>conv=</b>	Conversión de la entrada.
<b>seek=</b>	Número de bloques que hay que saltar al principio del fichero de salida.
<b>-s</b>	Shell (comando de conexión) por defecto del usuario (variable SHELL). El usuario puede cambiarlo mediante el comando <b>chsh</b> .
<b>-p</b>	La contraseña del usuario. ¡Cuidado! La contraseña debe estar ya cifrada. Así, a menos que se recopile la contraseña de una cuenta genérica, usted preferirá utilizar luego el comando <b>passwd</b> .

La opción **conv** admite los parámetros siguientes:

- **ascii**: convertir el EBCDIC en ASCII.
- **ebcdic**: convertir el ASCII en EBCDIC.
- **block**: completar los bloques que terminan por un salto de línea con espacios, hasta alcanzar el tamaño mencionado por **bs**.

- **unblock**: sustituir los espacios al final de bloques (de tamaño cbs) por un salto de línea.
- **lcase**: transformar las mayúsculas en minúsculas.
- **ucase**: transformar las minúsculas en mayúsculas.
- **noerror**: seguir incluso después de errores de lectura.
- **notrunc**: no limitar el tamaño del fichero de salida.
- **sync**: completar cada bloque leído con ceros para alcanzar el tamaño ibs.

Aquí colocará el sector de boot de la partición (donde está lilo o grub) en un fichero. El fichero así creado se podrá utilizar con el cargador de NT/2000/XP para un arranque en Linux.

```
# dd if=/dev/sdal of=boot.lnx bs=446 count=1
```

Para crear un fichero vacío con un tamaño de 1 MB:

```
$ dd if=/dev/zero of=vacio bs=1024  
count=1024  
1024+0 registros leídos  
1024+0 registros escritos  
1048576 bytes (1,0 MB) copied, 0,0199192 s, 52,6 MB/s
```

# El reloj

## 1. Conocer la hora

### a. fecha

Para conocer la hora, utilice el comando **date**. Permite dar la fecha actual, pero también calcular otras fechas en función o de la fecha actual o de cualquier fecha. **date** permite también modificar la fecha y la hora del sistema.

```
$ date  
sab may 10 13:58:38 CEST 2008
```

Por defecto, la fecha visualizada es la fecha (y hora) local, configurada en función del huso horario. Para visualizar la hora UTC:

```
$ date --utc  
sab may 10 11:01:10 UTC 2008
```

Se puede modificar el formato de la fecha a voluntad, tal como se puede hacer con la función C strftime. En este caso, la sintaxis es:

fecha +"formato"

Veamos algunos ejemplos de formato posibles:

Formato	Resultado
%H	La hora tiene el formato 00..23.
%M	Minutos 00..59.
%S	Segundos 00..60.
%T	Hora actual sobre 24 horas.
%r	Hora actual sobre 12 horas.
%Z	Huso horario.
%a	Día abreviado (lun, mar, etc.).
%A	Día completo.
%b	Mes abreviado.
%B	Mes completo.
%d	Día del mes.

<b>%j</b>	Día del año.
<b>%m</b>	Número del mes.
<b>%U</b>	Número de la semana 00..53.
<b>%y</b>	Dos últimas cifras del año.
<b>%Y</b>	Año completo.

Para visualizar una fecha completa:

```
$ date +"Hoy es %A, %d %B %Y, son las %H horas, %M minutos y %S segundos"
Hoy es sábado, 10 de mayo del 2008, son las 14 horas, 10 minutos y 20 segundos
```

Puede modificar la base de cálculo con el parámetro `--date` seguido de una fecha o de un cálculo. Se aceptan las palabras claves `today`, `yesterday`, `tomorrow`, `day(s)`, `week(s)`, `month(es)`, `year(s)`, `hour(s)`, `minute(s)`, `second(s)` con `+` (añadido a la fecha) o `-` o `ago` (quita a la fecha especificada). Si no se especifica la fecha, será la fecha en curso.

#### Dentro de 10 días:

```
date --date "10 days"
mar may 20 13:13:05 CEST 2008
```

#### Mañana:

```
date --date "tomorrow"
```

#### Ayer:

```
date --date "yesterday"
```

#### Una semana después de la Navidad de 2008:

```
date --date "12/25/2008 23:59:00 + 1 week"
jue ene 1 23:59:00 CET 2009
```

## b. hwclock

El comando **hwclock** permite interrogar directamente al reloj material RTC. El parámetro `--show` (por defecto) visualiza la fecha actual. Es diferente del tiempo del sistema que proviene de ntp o fecha. De hecho, al final se muestra un desajuste.

```
# hwclock --show
sáb 10 may 2008 13:16:14 CEST -0.390436 segundos
```

No es posible formatear el resultado del comando.

## 2. Modificar el reloj físico

Sólo se puede modificar el reloj físico como root mediante los comandos **date** (reloj del sistema interno al

núcleo) y **hwclock** (reloj físico).

### a. Mediante date

Modifique la fecha y la hora con el parámetro **-s**:

```
# date -s "05/09/2008 14:00"  
vie may 9 14:00:00 CEST 2008  
# date  
vie may 9 14:00:03 CEST 2008
```

### b. Mediante hwclock

El comando **hwclock** modifica el reloj físico (RTC) o el reloj del sistema. Como el reloj material es independiente de la hora del sistema, los resultados pueden sorprender.

```
# hwclock --set --date "05/10/2008 14:00"  
# date  
sáb may 10 13:30:53 CEST 2008  
# hwclock  
sáb 10 may 2008 14:00:13 CEST -0.658230 segundos
```

Usted puede sincronizar la hora del sistema y la hora física en los dos sentidos. Para que se sincronice la hora física desde la hora del sistema:

```
# hwclock --systohc  
# hwclock  
sáb 10 may 2008 13:34:00 CEST -0.931220 segundos
```

Para realizar lo contrario:

```
# hwclock --hctosys
```

## 3. NTP

### a. Fundamentos

**NTP** (*Network Time Protocol*) es un protocolo que permite sincronizar los relojes de los ordenadores mediante la red y, en particular, TCP/IP, o sea Internet. Como nuestros ordenadores utilizan relojes de cuarzo, a veces éstos se pueden adelantar o atrasar mucho dependiendo de la calidad de los componentes.

Existen numerosos ámbitos en los cuales no se puede admitir que un sistema no está a la hora, en particular por razones de sincronización muy precisas.

Un servidor NTP emite la hora en formato UTC. El cliente recupera la hora y la adapta en función de su huso horario. El servidor no gestiona tampoco los cambios de hora.

Si el servidor NTP está al día, la hora es muy precisa. Se codifica en 64 bits:

- los primeros 32 bits dan el número de segundos desde el 1º de enero de 1900 a medianoche (de manera que el bug NTP tendrá lugar antes del bug Unix);
- los últimos 32 bits dan la precisión de los segundos.

El nuevo protocolo NTP4 da una precisión de los segundos en 64 bits, lo que evita un inoportuno error

futuro.

Encontrará en la URL siguiente una lista de algunos servidores NTP españoles:

<http://www.bulma.net>

## b. Cliente ntp

El servicio ntpd permite sincronizar una máquina junto a un servidor de tiempo.

```
$ ps -ef|grep ntp
root      6523  5378  0 14:04 pts/2    00:00:00 ntpd
```

fichero de configuración es **/etc/ntp.conf**. En principio, este fichero ya contiene un determinado número de líneas que es preferible no tocar. Usted puede, incluso debe, añadir una línea que apunta al servidor de tiempo que ha elegido (por ejemplo es.pool.ntp.org):

```
server es.pool.ntp.org
```

Vuelva a iniciar el servicio. Su máquina debe ponerse en hora.

Es posible forzar una sincronización manual con el comando **ntpdate**. Este comando utiliza como parámetro un nombre de servidor ntp.

```
# ntpdate es.pool.ntp.org
10 May 14:09:21 ntpdate[6551]: adjust time server 193.54.82.20 offset 0.154057 sec
```

Si no desea utilizar el servicio ntpd, puede colocar este comando en crontab todos los días o todas las horas.

# Los parámetros regionales

## 1. i18n y I10n

Las distribuciones se instalan y funcionan en numerosos idiomas. Seguramente habrá notado que las páginas de los manuales se muestran en su idioma si se han traducido. Del mismo modo, un gran número de programas muestran su menú en el idioma en el cual se instaló o configuró el sistema.

A diferencia de numerosos editores de programas que proveen versiones localizadas diferenciadas de sus programas (por ejemplo, MS Office en español y en inglés son dos versiones distintas), los editores de programas libres suelen integrar directamente un soporte para numerosos idiomas o facilitan paquetes adicionales.

Hay que diferenciar entre la regionalización (en inglés, «localization») y la internacionalización:

- La **regionalización** consiste en facilitar una traducción de un producto que corresponde a la cultura local (idioma principalmente, pero también moneda o representación de los números). A título de ejemplo, si bien el castellano se utiliza en España, México, Argentina y otros países de América Latina, cada cultura dispone de sintaxis y giros que pueden cambiar, así como de monedas diferentes.
- La **internacionalización** prepara la regionalización antes, a nivel de desarrollo del programa. Un solo programa requerirá funciones de una API de internacionalización. Una función encargada de recuperar una cadena de texto en cualquier idioma dentro de una base verificará qué idioma se utiliza y sacará la cadena correspondiente. La librería de funciones `gettext` es un buen ejemplo.

En vez de utilizar un modelo fijado como éste:

```
printf("Hello");
```

un programador podrá proceder como a continuación:

```
printf(gettext("Hello"));
```

La función `gettext` buscará la cadena correspondiente a "Hello" en los ficheros de regionalización del programa en el idioma actual y sacará una versión traducida. En español, la segunda línea será:

Hola

---

 Para ser precisos, internacionalización se suele escribir i18n (i seguido de 18 letras y luego n), y regionalización, I10n (I para localization, seguido de 10 letras y luego n).

---

## 2. Configuraciones locales

### a. Herramientas de la distribución

Hay varias maneras de modificar la regionalización del sistema, incluso de un programa dado. Cada distribución provee un módulo de configuración. Por ejemplo, openSUSE facilita dentro de YaST un módulo de gestión del idioma y lo mismo se puede hacer con Debian de la manera siguiente:

```
dpkg-reconfigure locales
```

Estos métodos sólo modifican parámetros de sistema fácilmente accesibles. Es perfectamente posible modificarlos de forma manual mediante los ficheros de configuración de texto. Del mismo modo, cada

entorno de escritorio facilita herramientas de regionalización mediante su centro de configuración. Es posible, para un programa dado, pasar de uno a otro.

## b. Variables de entorno

Con la consola o en el entorno de escritorio, si se deja por defecto la regionalización, se recupera la información utilizada relativa al país, el idioma o la moneda gracias a unas variables de entorno del shell. La principal se llama **LANG**. Se suele colocar esta variable en /etc/profile u otros ficheros (/etc/sysconfig/language en openSUSE, por ejemplo). Su contenido en el puesto del autor es el siguiente:

```
LANG=es_ES.UTF-8
```

Su formato es el siguiente:

```
idioma_Pais[.norma][@variante]
```

- **idioma** es el idioma utilizado (aquí es, para España).
- **País** es el país en mayúscula (aquí ES).
- **.norma** indica la norma utilizada para la codificación de los caracteres (aquí en UTF-8).
- **@variante** precisa variante del idioma. En algunos sistemas que utilizan la codificación ISO-8859-15 se puede especificar @euro para indicar una variante que soporta el euro de esta codificación.

Para pasar al inglés americano:

```
export LANG=en_US.UTF-8
```

El comando **locale** permite recuperar información sobre los elementos de regionalización soportados por su sistema:

```
seb@slyserver:/etc/sysconfig> locale -a|grep es
es_MX
es_MX@euro
es_MX.utf8
es_AR
es_AR.utf8
fr_CL
fr_CL.utf8
es_ES
es_ES@euro
es_ES.utf8
fr_PE
fr_PE@euro
fr_PE.utf8
```

Los elementos de regionalización soportados por el sistema están en **/usr/share/locale**.

La variable LANG sustituye por defecto a las demás variables de entorno de regionalización, excepto si existen a un nivel inferior. En numerosos Unix, entre los cuales se halla Linux, existe un gran número de variables que empiezan con LC\_. Si LANG es la única variable, sustituye a todas estas variables.

Si llama al comando **locale** sin parámetro, muestra lo siguiente:

```
seb@slyserver:/etc/sysconfig> locale
LANG=es_ES.UTF-8
LC_CTYPE="es_ES.UTF-8"
LC_NUMERIC="es_ES.UTF-8"
```

```
LC_TIME="es_ES.UTF-8"
LC_COLLATE="es_ES.UTF-8"
LC_MONETARY="es_ES.UTF-8"
LC_MESSAGES="es_ES.UTF-8"
LC_PAPER="es_ES.UTF-8"
LC_NAME="es_ES.UTF-8"
LC_ADDRESS="es_ES.UTF-8"
LC_TELEPHONE="es_ES.UTF-8"
LC_ALL=
```

Se puede modificar y adaptar cada una de las variables LC. Veamos su significado:

- LC\_CTYPE: clase de caracteres y conversión.
- LC\_NUMERIC: formato numérico por defecto, diferente del de la moneda.
- LC\_TIME: formato por defecto de la fecha y la hora.
- LC\_COLLATE: reglas de comparación y de ordenación (por ejemplo, los caracteres acentuados).
- LC\_MONETARY: formato monetario.
- LC\_MESSAGES: formato de los mensajes informativos, interactivos y de diagnóstico.
- LC\_PAPER: formato de papel por defecto (por ejemplo, A4).
- LC\_NAME: formato del nombre de una persona.
- LC\_ADDRESS: ídem para una dirección.
- LC\_TELEPHONE: ídem para el teléfono.
- LC\_ALL: reglas para todas las otras variables LC.

Veamos un ejemplo con una modificación de LC\_TIME:

```
seb@slyserver:~> date
vie. may  8 11:28:36 CEST 2009
seb@slyserver:~> export LC_TIME=en_US
seb@slyserver:~> date
Fri May  8 11:28:52 CEST 2009
```

 En algunos casos, es necesario exportar la variable LANG=C dentro de un script por ejemplo. Eso tiene como efecto restablecer la regionalización de los comandos, en general en inglés, o de tipo POSIX, con el fin de asegurar un funcionamiento óptimo de los comandos correspondientes.

### 3. Codificación de los caracteres

No todo el mundo utiliza el alfabeto occidental. E incluso en los países occidentales, existen numerosas variantes de un mismo alfabeto. Los ingleses no utilizan acentos, los alemanes tienen caracteres adicionales y otros países emplean más signos auxiliares. Es aún más complicado con los países que no utilizan el mismo alfabeto o que usan ideogramas.

Los nombres de los ficheros pueden ser ilegibles en otro país. El contenido de texto de un fichero, también.

Cada variante necesita la aplicación de una variante para la codificación de los caracteres. Durante mucho tiempo, la tabla se conocía como ASCII. Se trata de una tabla de caracteres codificada en 1 byte. Por lo tanto, esta tabla puede contener 256 caracteres. Como los primeros 127 son invariables, contiene todos los caracteres americanos, o sea, ningún carácter acentuado, así como los caracteres de control y formateo (supresión, retorno de carro, etc.). Los siguientes 128 son libres. En España se colocan caracteres

semigráficos y los acentos, y lo mismo en otros países. Esto significa que, para cada país, hay que cambiar la tabla ASCII. Además, si usted accede a una palabra o un nombre de fichero creado en un país y que contiene caracteres propios de este país, hay un riesgo importante de que los nombres ya no se correspondan.

Estas tablas se han normalizado con las normas ISO. Por ejemplo, para España, la tabla ISO 8859-1 corresponde a los caracteres de Europa del Oeste en Unix; ISO 8859-15 después del paso al euro. En Windows es la tabla Windows-1252; en DOS, CP850, etc. Cada tabla es incompatible con las demás. Se deben adaptar las fuentes de caracteres para cada tabla. Pero lo peor es que los 256 caracteres no bastan para codificar todos los ideogramas.

Para remediar este problema, existe una norma informática llamada **unicode** compatible con el estándar ISO 10646: da a cualquier carácter de cualquier sistema de escritura de idioma, un nombre y un identificador numérico, y eso de manera unificada, sea cual sea la plataforma informática o el programa (definición inspirada de Wikipedia). Si un sistema acepta unicode y dispone de las fuentes de caracteres asociadas, entonces se visualizan los textos y los nombres de ficheros correctamente, sea cual sea el idioma y los caracteres utilizados. Se podrá visualizar correctamente un nombre de fichero en chino en un sistema en francés.

Linux, así como numerosas páginas de Internet, utilizan el formato unicode UTF-8 por defecto. Ahora, los sistemas más corrientes gestionan correctamente este formato.

Es posible convertir un fichero codificado en una tabla dada hacia otra tabla con el programa **iconv**. El parámetro **-l** le da todas las tablas soportadas. La salida está truncada, ya que ocuparía 4 páginas.

```
slyserver:/usr/share/zoneinfo # iconv -l
...
437, 500, 500V1, 850, 851, 852, 855, 856, 857, 860, 861, 862, 863,
864, 865, 866, 866NAV, 869, 874, 904, 1026, 1046, 1047, 8859_1,
8859_2, 8859_3, 8859_4, 8859_5, 8859_6, 8859_7, 8859_8, 8859_9,
10646-1:1993, 10646-1:1993/UCS4,
...
IS0646-ES2, IS0646-FI, IS0646-FR, IS0646-FR1, IS0646-GB, IS0646-HU,
IS0646-IT, IS0646-JP-OCR-B, IS0646-JP, IS0646-KR, IS0646-N0, IS0646-N02,
...
WINDOWS-31J, WINDOWS-874, WINDOWS-936, WINDOWS-1250, WINDOWS-1251,
WINDOWS-1252, WINDOWS-1253, WINDOWS-1254, WINDOWS-1255, WINDOWS-1256,
WINDOWS-1257, WINDOWS-1258, WINSAMI2, WS2, YU
```

Para convertir un fichero, utilice la sintaxis siguiente:

```
iconv -f WINDOWS-1252 -f UTF8 nombre_fichero
```

## 4. Husos horarios

Linux gestiona los husos horarios. Por defecto, sin configuración de huso, se elige la hora universal UTC, tiempo universal coordinado, invariable en el mundo. El desfase de los husos horarios se efectúa en relación con el tiempo UTC.

En Debian, se coloca la definición del huso horario actual en **/etc/timezone**, y en Redhat (o derivado), en **/etc/localtime**. A menudo son vínculos (atajos) o una copia de la entrada correspondiente en **/usr/share/zoneinfo**.

Aunque una vez más las herramientas de la distribución permitan elegir su huso, usted puede modificarlo manualmente con el comando **tzselect**. Linux gestiona automáticamente el cambio a la hora de verano o invierno.

# Validación de los conocimientos adquiridos: preguntas/respuestas

## 1. Preguntas

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **Gestión de los usuarios y grupos**

**1** ¿Permite un UID autenticar un usuario?

**2** ¿Puede una persona formar parte de varios grupos?

**3** ¿Debe dar el UID 85 a un usuario simple del sistema?

**4** ¿Con qué método se autentica un usuario en Linux?

- A - Con su login.
- B - Con su UID.
- C - Con su contraseña.
- D - Con una clave secreta.

**5** ¿Qué comando permite conocer los UID y GID de un usuario?

**6** ¿Cuál es el camino completo del fichero que contiene las contraseñas cifradas?

**7** ¿Cuál es el cuarto campo de una línea del fichero **/etc/passwd**?

- A - El UID.
- B - El GID.
- C - El comentario.
- D - El directorio de trabajo.

**8** ¿Cuál es el cuarto campo del fichero **/etc/group**?

- A - El GID.
- B - La contraseña de grupo.
- C - La lista de los usuarios que tiene este grupo como grupo primario.
- D - La lista de los usuarios que tiene este grupo como grupo secundario.

**9** ¿Puede root conocer la contraseña de un usuario?

**10** La línea `useradd -m -u 100 -g users -G video -c test -s /bin/false`:

- A - No funciona.
- B - Añade un usuario prueba de UID 100.
- C - Añade un usuario a video pero que no tiene derecho a conectarse.
- D - Crea un usuario de grupo secundario video.

**11** Para añadir un usuario a un grupo usted puede:

- A - Utilizar usermod -G.
- B - Utilizar groupmod -A.
- C - Utilizar vigr.

- D - Utilizar vi /etc/group.

**12** ¿Qué comando se debe insertar para forzar al usuario a cambiar su contraseña cada 30 días?

**13** El usuario joe ha sido despedido de la empresa. ¿Cómo bloquear su cuenta?

**14** Joe tenía documentos importantes en su directorio personal. ¿Cómo suprimir su cuenta sin suprimir su carpeta personal?

**15** ¿Qué comando puede utilizar para comprobar el estado de sus ficheros de usuarios y de contraseñas?

**16** Acaba de migrar de un servidor Unix, que no gestionaba los shadows, hacia Linux. Para migrar las contraseñas, debe:

- A - Utilizar pwconv.
- B - Utilizar grpconv.
- C - Crear un script de creación de /etc/shadow.
- D - No hacer nada: Linux no tiene por qué utilizar los shadows.

**17** ¿Cómo saber cuándo se conectó jack por última vez?

**18** Un usuario intenta cambiar su shell en /bin/zsh (que existe) con chsh, pero no lo consigue. ¿Qué tiene que comprobar primero?

**19** El usuario william intenta obtener clase1 como grupo primario el tiempo de su conexión, pero no lo consigue con el comando newgrp. ¿Por qué?

**20** Debe adoptar de manera temporal los derechos de root para lanzar un servicio. Usted conoce su contraseña. Debe:

- A - Desconectar y abrir una sesión root.
- B - Utilizar el comando **su**.
- C - Trabajar en root directamente.
- D - Utilizar el comando **login**.

**21** ¿Qué fichero debe modificar para que todos los usuarios dispongan tras su creación del shell /bin/ksh?

**22** ¿Qué variable del fichero /etc/login.defs debe modificar para que todos los UID de los usuarios se inicien a 1500?

**23** Dé el orden de visualización de los mensajes a la conexión local:

- A - /etc/issue.net luego /etc/motd.
- B - /etc/motd luego /etc/issue.
- C - /etc/issue.net, /etc/issue luego /etc/motd.
- D - /etc/issue luego /etc/motd.

**24** ¿Qué directorio por defecto contiene el perfil de un usuario genérico?

**25** ¿Qué fichero debe crear para prohibir cualquier conexión en un servidor Linux que utiliza los PAM?

## Impresión

**26** ¿Qué nombre del sistema de impresión se suele utilizar en Linux?

- A - BSD
- B - System V
- C - CUPS
- D - LPD

**27** Con los comandos System V, ¿cómo imprimir un fichero de texto llamado fi.txt en la fila de impresión

Ij2100?

- 28** Ya sea en System V o BSD, ¿cuál es el nombre del servicio de impresión?
- 29** En este mismo caso, dé la ruta del fichero de configuración de las colas de impresión.
- 30** ¿Qué comandos System V y BSD funcionan con CUPS?
- 31** ¿Es aconsejable modificar el fichero `/etc/cups/printers.conf` manualmente?
- 32** ¿Cuál es la URL de configuración de CUPS en la máquina local?
- 33** ¿Es necesario declarar varias filas para una impresora si desea modificar las opciones de impresión para cada una de las filas?

### **Automatización**

- 34** ¿Cuál es el efecto de la línea crontab siguiente \* \* \* \* \* who > /tmp/test?
- 35** ¿Qué línea crontab se debe escribir para ejecutar el comando **checkall** de lunes a viernes a las 23 h, únicamente en el mes de junio?
- A - \* 23 \* 6 1,5 checkall
  - B - 0 23 0 6 1-5 checkall
  - C - 23 \* 6 1-5 checkall
  - D - 0 23 \* 6 1-5 checkall
- 36** Dé la ruta de la configuración crontab del sistema.
- 37** ¿Qué comando permite editar el crontab del usuario pedro?
- 38** Excepcionalmente debe iniciar una tarea llamada script.sh este noche a las 21:00 h desde su consola. ¿Qué comando debe teclear?
- 39** Se ha cancelado la tarea. ¿Cómo listar y, luego, suprimir la tarea?
- 40** ¿Cómo prohibir a un usuario utilizar el crontab?

### **Syslog**

- 41** ¿Qué nombre lleva el servicio que traza la información emitida por el núcleo?
- A - syslogd.
  - B - klogd.
  - C - kerneld.
  - D - dmesg.
- 42** ¿Puede syslogd tratar los mensajes emitidos por el núcleo?
- 43** Dé la ruta del fichero de configuración de syslog.
- 44** syslog contiene una línea \*.\*/var/log/messages. ¿Qué contiene /var/log/messages?
- 45** Desea que todos los mensajes relacionados con la seguridad de los usuarios lleguen a /var/log/secure. ¿Qué línea debe añadir en el fichero de configuración de syslog?

### **Archivado**

- 46** ¿Es pertinente efectuar una copia de seguridad completa todos los días?
- 47** Para usted, ¿cuál es el método más fiable de archivado?
- A - En un CD/DVD.
  - B - En un disco duro externo.

- C - En una llave USB.
- D - En otra partición del disco duro.

**48** ¿Cómo archivar y comprimir al formato bzip2 el directorio /datos en el archivo datos.tar.bz2?

**49** ¿Cómo restaurar un archivo cpio llamado archivo.cpio con todos sus subdirectorios, pero sin chafar los ficheros ya presentes?

### **Reloj**

**50** Dé dos métodos para conocer la hora actual.

**51** El 2 de diciembre de 1975 era un:

- A - lunes
- B - martes
- C - miércoles
- D - jueves

**52** ¿Cuáles son los métodos válidos para poner en hora su máquina?

- A - hwclock --set
- B - date -s
- C - ntpdate servidor\_ntp.com
- D - hwclock --systohc

**53** Dé la ruta del fichero de configuración del cliente NTP.

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /53

Para este capítulo, su resultado mínimo debe ser de 40 respuestas acertadas (40/53).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- Gestión de los usuarios y grupos.
- Impresión.
- Automatización.
- Syslog.
- Archivado.
- Reloj.

## **3. Respuestas**

### **Gestión de los usuarios y grupos**

**1** ¿Permite un UID autenticar un usuario?

No. El UID identifica a la persona.

**2** ¿Puede una persona formar parte de varios grupos?

*Sí. La persona dispondrá de todos los derechos asociados a cada uno de los grupos.*

**3** ¿Debe dar el UID 85 a un usuario simple del sistema?

*No. Se reservan los UID inferiores a 100 para los usuarios con privilegios.*

**4** ¿Con qué método se autentica un usuario en Linux?

- A - Con su login.
- B - Con su UID.
- C - Con su contraseña.
- D - Con una clave secreta.

*C. El usuario se autentica con la contraseña, que sólo él conoce.*

**5** ¿Qué comando permite conocer los UID y GID de un usuario?

*id.*

**6** ¿Cuál es el camino completo del fichero que contiene las contraseñas cifradas?

*/etc/shadow*

**7** ¿Cuál es el cuarto campo de una línea del fichero /etc/passwd?

- A - El UID.
- B - El GID.
- C - El comentario.
- D - El directorio de trabajo.

*B. Es el GID del grupo primario.*

**8** ¿Cuál es el cuarto campo del fichero /etc/group?

- A - El GID.
- B - La contraseña de grupo.
- C - La lista de los usuarios que tienen este grupo como grupo primario.
- D - La lista de los usuarios que tienen este grupo como grupo secundario.

*D.*

**9** ¿Puede root conocer la contraseña de un usuario?

*No, está encriptada. Pero no la necesita.*

**10** La línea `useradd -m -u 100 -g users -G vídeo -c test -s /bin/false`:

- A - No funciona.
- B - Añade un usuario prueba de UID 100.
- C - Añade un usuario a vídeo pero que no tiene derecho a conectarse.
- D - Crea un usuario de grupo secundario vídeo.

A. Falta el nombre del usuario.

**11** Para añadir un usuario en un grupo usted puede:

- A - Utilizar usermod -G.
- B - Utilizar groupmod -A.
- C - Utilizar vigr.
- D - Utilizar vi /etc/group.

*Todas las respuestas son correctas!*

**12** ¿Qué comando se debe insertar para forzar al usuario a cambiar su contraseña cada 30 días?

*chage -M 30 bob o passwd -x 30 bob.*

**13** El usuario joe ha sido despedido de la empresa. ¿Cómo bloquear su cuenta?

*Con passwd -l joe o usermod -L joe.*

**14** Joe tenía documentos importantes en su directorio personal. ¿Cómo suprimir su cuenta sin suprimir su carpeta personal?

*userdel joe sin la -r.*

**15** ¿Qué comando puede utilizar para comprobar el estado de sus ficheros de usuarios y de contraseñas?

*El comando pwck.*

**16** Acaba de migrar de un servidor Unix, que no gestionaba los shadows, hacia Linux. Para migrar las contraseñas, debe:

- A - Utilizar pwconv.
- B - Utilizar grpconv.
- C - Crear un script de creación de /etc/shadow.
- D - No hacer nada: Linux no tiene por qué utilizar los shadows.

*A. pwconv convertirá /etc/passwd y verterá su contenido en dos ficheros, un de ellos /etc/shadow. Observe que la solución D también es correcta, pero no idónea.*

**17** ¿Cómo saber cuándo se conectó jack por última vez?

*lastlog -u jack o last jack.*

**18** Un usuario intenta cambiar su shell en /bin/zsh (que existe) con chsh, pero no lo consigue. ¿Qué tiene que comprobar primero?

*¿chsh -l muestra /bin/zsh? Si no es el caso, añada este shell en /etc/shells después de comprobar que haya sido instalado correctamente. En efecto, los scripts de postinstalación de los paquetes añaden en principio las entradas correspondientes a este archivo.*

**19** El usuario william intenta obtener clase1 como grupo primario el tiempo de su conexión, pero no lo consigue con el comando **newgrp**. ¿Por qué?

*Es más que probable que william no forme parte del grupo clase1. Verifique si forma parte de los grupos secundarios con id.*

**20** Debe adoptar de manera temporal los derechos de root para iniciar un servicio. Usted conoce su contraseña.

Debe:

- A - Desconectar y abrir una sesión root.
- B - Utilizar el comando **su**.
- C - Trabajar en root directamente.
- D - Utilizar el comando **login**.

*B. No trabajar nunca como root de forma permanente.*

**21** ¿Qué fichero debe modificar para que todos los usuarios dispongan tras su creación del shell /bin/ksh?

*El fichero /etc/default/useradd, línea SHELL.*

**22** ¿Qué variable del fichero /etc/login.defs debe modificar para que todos los UID de los usuarios se inicien a 1500?

*UID\_MIN debe pasar a 1500.*

**23** Dé el orden de visualización de los mensajes a la conexión local:

- A - /etc/issue.net luego /etc/motd.
- B - /etc/motd luego /etc/issue.
- C - /etc/issue.net, /etc/issue luego /etc/motd.
- D - /etc/issue luego /etc/motd.

*D. Se trata de una conexión local, por lo tanto no se utiliza /etc/issue.net.*

**24** ¿Qué directorio por defecto contiene el perfil de un usuario genérico?

*/etc/skel*

**25** ¿Qué fichero debe crear para prohibir cualquier conexión en un servidor Linux que utiliza los PAM?

*/etc/nologin.*

### **Impresión**

**26** ¿Qué nombre del sistema de impresión se suele utilizar en Linux?

- A - BSD
- B - System V
- C - CUPS
- D - LPD

*C: CUPS, Common Unix Printing System.*

**27** Con los comandos System V, ¿cómo imprimir un fichero de texto llamado fi.txt en la fila de impresión lj2100?

*lpr -Plj2100 fic.txt*

**28** Ya sea en System V o BSD, ¿cuál es el nombre del servicio de impresión?

*lpd.*

**29** En este mismo caso, dé la ruta del fichero de configuración de las filas de impresión.

*/etc/printcap.*

**30** ¿Qué comandos System V y BSD funcionan con CUPS?

*Todos. CUPS asegura una compatibilidad en los dos casos.*

**31** ¿Es aconsejable modificar el fichero /etc/cups/printers.conf manualmente?

*No, es preferible pasar por la interfaz HTML o por una herramienta gráfica.*

**32** ¿Cuál es la URL de configuración de CUPS en la máquina local?

*http://localhost:631.*

**33** ¿Es necesario declarar varias colas para una impresora si desea modificar las opciones de impresión para cada una de las colas?

*No. Era el caso con LPD, donde cada fila tenía sus propiedades y no era posible cambiar fácilmente los parámetros de impresión a voluntad. Por supuesto, esto aún se produce, pero las API de CUPS permiten la creación de cuadros de diálogo de propiedades de impresión por fila, como en Windows.*

### **Automatización**

**34** ¿Cuál es el efecto de la línea crontab siguiente \* \* \* \* \* who > /tmp/test?

*Cada minuto, el resultado del comando who chafa el antiguo contenido de /tmp/test, la lista de los usuarios conectados localmente.*

**35** ¿Qué línea crontab se debe escribir para ejecutar el comando **checkall** de lunes a viernes a las 23 h, únicamente en el mes de junio?

- A - \* 23 \* 6 1,5 checkall
- B - 0 23 0 6 1-5 checkall
- C - 23 \* 6 1-5 checkall
- D - 0 23 \* 6 1-5 checkall

*D. 0 23: 23h 00, \* 6 1-5. En junio, de lunes a viernes.*

**36** Dé la ruta de la configuración crontab del sistema.

*/etc/crontab.*

**37** ¿Qué comando permite editar el crontab del usuario pedro?

*crontab -u pedro*

**38** Excepcionalmente debe iniciar una tarea llamada script.sh este noche a las 21:00 h desde su consola. ¿Qué comando debe teclear?

*at -f script.sh 21:00*

**39** Se ha cancelado la tarea. ¿Cómo listar y, luego, suprimir la tarea?

*atq para listar, y atrm seguido del número de job para suprimirla.*

**40** ¿Cómo prohibir a un usuario utilizar el crontab?

*Añadiendo su login en /etc/cron.deny.*

## **Syslog**

**41** ¿Qué nombre lleva el servicio que traza la información emitida por el núcleo?

- A - syslogd.
- B - klogd.
- C - kerneld.
- D - dmesg.

*B. klogd traza los mensajes emitidos por el núcleo.*

**42** ¿Puede syslogd tratar los mensajes emitidos por el núcleo?

*Sí. klogd vuelve a formatear y mandar los mensajes a syslogd.*

**43** Dé la ruta del fichero de configuración de syslog.

*/etc/syslog.conf.*

**44** syslog contiene una línea \*.\* /var/log/messages. ¿Qué contiene /var/log/messages?

*/var/log/messages contiene la totalidad de los mensajes emitidos por todos los componentes (núcleo, servicios y programas) que utilizan syslog.*

**45** Desea que todos los mensajes relacionados con la seguridad de los usuarios lleguen a /var/log/secure. ¿Qué línea debe añadir en el fichero de configuración de syslog?

*auth.\* /var/log/secure*

## **Archivado**

**46** ¿Es pertinente efectuar una copia de seguridad completa todos los días?

*En principio, no. Puede establecer copias de seguridad completas a intervalos regulares y luego efectuar una copia incremental entre las dos copias completas.*

**47** Para usted, ¿cuál es el método más fiable de archivado?

- A - En un CD/DVD.
- B - En un disco duro externo.
- C - En una llave USB.
- D - En otra partición del disco duro.

*A. Los otros soportes no son perennes y en cualquier caso están sometidos a más riesgos.*

**48** ¿Cómo archivar y comprimir al formato bzip2 el directorio /datos en el archivo datos.tar.bz2?

*tar cvjf datos.tar.bz2 /datos*

**49** ¿Cómo restaurar un archivo cpio llamado archivo.cpio con todos sus subdirectorios, pero sin chafar los ficheros ya presentes?

*cat archivo.cpio |cpio -d*

## **Reloj**

**50** Dé dos métodos para conocer la hora actual.

`date y hwclock -show.`

**51** El 2 de diciembre de 1975 era:

- A - lunes
- B - martes
- C - miércoles
- D - jueves

`date --date "12/02/1975" dice que era martes.`

**52** ¿Cuáles son los métodos válidos para poner en hora su máquina?

- A - `hwclock --set`
- B - `date -s`
- C - `ntpdate servidor_ntp.com`
- D - `hwclock --systohc`

*A, B y C son correctas para poner el sistema en hora según una fecha especificada, mientras que D es correcta si el sistema está ya en hora y lo sincroniza con el reloj material.*

**53** Dé la ruta del fichero de configuración del cliente NTP.

`/etc/ntp.conf`

# Requisitos y objetivos

## 1. Requisitos

- Tener conocimientos básicos relativos al funcionamiento de una red.
- Disponer de una conexión Ethernet.
- Si es posible, disponer de una segunda máquina o poder acceder a otras máquinas de la red.
- Tener privilegios de root.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Comprender, trabajar con los protocolos TCP/IP.
- Configurar las interfaces de red.
- Utilizar los principales comandos.
- Configurar el resolver y los principales ficheros.
- Implementar los servicios xinetd.
- Establecer una conexión PPP mediante un módem.
- Utilizar los clientes y servidores SSH.
- Configurar y consultar un DNS.
- Configurar un servidor de correo electrónico básico.
- Configurar POP e IMAP.
- Crear un servidor FTP.
- Montar un servidor Apache básico.
- Compartir sus ficheros con NFS y Samba.

# Ficheros compartidos

## 1. NFS

### a. Inicio

El sistema de ficheros compartidos **NFS** (*Network File System*) o sistema de ficheros de red permite compartir todo o parte de un sistema de ficheros destinados a clientes NFS, a menudo otros Unix. En su versión básica, es un sistema simple y eficaz. Vamos a estudiar la versión 2.

NFS se apoya en el **portmapper** (**portmap**), el soporte **nfs** del núcleo y los servicios **rpc.nfssd** y **rpc.mountd**.

Para iniciar el servicio NFS, se debe ejecutar portmap y nf (verificar el estado antes).

```
# service portmap status # /etc/init.d/portmap status o rpcinfo -p  
# service nfs status  
# service portmap start  
# service nfs start  
# service nfslock start
```

Para saber si el servicio está disponible en un anfitrión remoto:

```
# rpcinfo -p anfitrión
```

### b. Compartir por el lado servidor

La lista de directorios que hay que exportar se encuentra en **/etc/exports**. Contiene un sistema compartido por línea.

```
# Dir exports Autorizaciones de acceso  
/ puesto1(rw) puesto2(rw,no_root_squash)  
/projects *.midominio.org(rw)  
/home/joe puesto*.midominio.org(rw)  
/pub 192.168.1.0/255.255.255.0(ro)
```

Cada línea se compone de dos partes. La primera es la ruta del directorio exportado. La segunda contiene las autorizaciones de acceso.

La autorización de acceso se compone de pares huésped/permisos según el formato siguiente:

huésped(permisos)

Si no se define el huésped, es a toda la red (de alcance mundial) a la que se aplicarán los permisos. Si no se definen los permisos, la exportación será de sólo lectura. Sobre todo, no debe ponerse espacio entre el huésped y los permisos. El huésped puede ser:

- un nombre de huésped único;
- un dominio;
- una red o subred;
- una combinación del conjunto con caracteres de sustitución (\*, ?).

Los permisos pueden ser:

- **ro**: sólo lectura;
- **rw**: lectura escritura;
- **no\_root\_squash**: el root remoto equivale al root local;
- **root\_squash**: si root se conecta al directorio compartido, su uid será reemplazado por el de un usuario anónimo. Así no hay riesgo de que el usuario root de un puesto local pueda ser root en un directorio remoto;
- **all\_squash**: extiende la regla precedente a todos los usuarios;
- **anonuid / anongid**: uid y gid para el usuario anónimo.

Para una gestión correcta de los derechos y permisos, **los usuarios de mismo nombre (login) deben tener los mismos UID y GID en el servidor y el cliente**. NFS se basa, en efecto, en estos valores para garantizar la seguridad de los datos del directorio compartido. El nombre de login solo no es suficiente. En el ejemplo siguiente, el usuario **joe** está autorizado a acceder al directorio compartido/**home/joe** (se supone que es su directorio personal) en todos los puestos del dominio. Se debe declarar al usuario joe de la misma manera (mismo UID) en el servidor y en todos los puestos. Por eso se utiliza a menudo NIS con NFS.

El comando **exportfs** permite controlar los directorios compartidos.

- **exportfs -r**: refresca la lista de directorios compartidos después de la modificación de /etc/exports;
- **exportfs -v**: lista de los directorios compartidos;
- **exportfs -a**: exporta (o vuelve a cargar) todos los directorios compartidos de /etc/exports o un directorio compartido dado;
- **exportfs -u**: detiene la compartición dada. **-a** para todos.

El comando **showmount** muestra los directorios compartidos de un anfitrión dado.

```
showmount -e host
```

### c. Montaje por el lado cliente

El soporte NFS está incluido en forma de módulo del núcleo. Se carga automáticamente cuando se utiliza un acceso NFS.

En **/etc/fstab** observe las modificaciones:

```
server1:/pub /mnt/pub nfs defaults 0 0
```

El periférico se ha reemplazado por la ruta del directorio compartido con la forma **servidor:ruta**. El sistema de ficheros es **nfs**. Es lo mismo con el comando **mount**:

```
mount -t nfs servidor1:/pub /mnt/pub
```

---

 Si se definen los montajes NFS en /etc/fstab, mount -a no los va a montar. Un servicio suele estar presente en cada distribución para montarlos y desmontarlos en las paradas/reinicios. En Red Hat, el servicio /etc/rc.d/init.d/netfs los montará de manera automática durante el inicio.

---

NFS dispone de opciones de montaje específicas:

- **nolock**: opción de compatibilidad con antiguos servidores NFS;

- **intr**: interrumpe una petición NFS si el servidor no contesta;
- **hard**: bloquea los procesos que intentan acceder a un directorio compartido inaccesible;
- **soft**: un proceso devolverá un error en caso de acceso infructuoso;
- **rsize=8192, wsize=8192**: tamaño de los bloques de lectura/escritura en el servidor. Una escritura de 8 kB es más rápida que 8 escrituras de 1 kB.

¡Atención al tipo de montaje NFS! Durante mucho tiempo, incluidas algunas distribuciones basadas en el núcleo 2.6, los montajes NFS utilizaban el protocolo UDP. Sin embargo, esta opción de montaje se ha modificado y ahora se utiliza TCP. En UDP, cuando una transmisión fallaba, saltaba un timeout. Más allá de un valor determinado, significa que se tendrá que retransmitir un paquete RPC completo con cada error de transmisión, siendo bastante goloso.

En TCP, el control de transmisión se deja en manos del propio protocolo para los paquetes más pequeños. Ahora bien, esto puede provocar efectos secundarios importantes cuando el servidor está trabajando con demasiada carga o cuando la entrada/salida es demasiado grande en un montaje, ya que el valor de timeout por defecto no ha cambiado. Entonces, puede ser que suceda lo siguiente: un archivo se escribe en el montaje NFS; por una razón cualquiera, el paquete TCP no llega en el tiempo previsto (por unas décimas de segundo) y NFS solicita la retransmisión de los datos sin esperar el timeout TCP, lo que provoca un conflicto, una corrupción de datos y la aparición de mensajes de error:

```
nfs: server not responding, timed out
```

o incluso:

```
touch: cannot touch `/home/seb/test': Input/output error
```

Hay que modificar la opción de montaje dejando TCP pero modificando el timeout NFS, expresado en décimas de segundo, para hacerlo corresponder al de TCP. Utilice un valor de 600, es decir, 60 segundos. En todas las últimas versiones de NFS para Linux, se ha convertido en el valor predeterminado, pero para estar seguro de que no se encontrará con este problema, utilice el comando como en el ejemplo siguiente:

```
mount -o timeo=600,retrans=2, tcp -t nfs servidor1:/pub /mnt/pub
```

## FTP

El servidor **FTP** (*File Transfer Protocol*) más habitual es **vsftpd** (*Very Secure FTP Daemon*). Tiene la ventaja de ser muy pequeño, eficaz y rápido y al mismo tiempo muy configurable (sin embargo, menos que Proftpd u otros). Es adecuado para la casi totalidad de las situaciones. Se trata de un servicio que se puede iniciar tanto por **xinetd** como (en las últimas versiones de las distribuciones) a título de servicio individual.

Hay dos niveles de seguridad a disposición del usuario:

- **Anonyme**: todo el mundo se puede conectar al servidor FTP como usuario **ftp** o **anonymous**. El entorno FTP está chrooteado.
- **Usuario**: los usuarios que existen en el servidor pueden conectarse con su contraseña y tienen un acceso completo a sus datos en su directorio personal.

Dado que los usuarios anónimos se consideran como el usuario ftp, es el directorio personal de esta cuenta el que constituye la raíz del ftp.

El fichero de configuración se encuentra en `/etc/vsftpd/vsftpd.conf`.

La raíz del ftp por defecto está en `/var/ftp`.

El script de ejecución es `/etc/init.d/vsftpd` (servicio `vsftpd start`).

Para activar o no el acceso anónimo, se modifica el fichero de configuración. En este caso, el usuario se puede conectar como anonymous o ftp. En todos los casos, se reconocerá como usuario «ftp» del servidor una vez conectado:

```
anonymous_enable=YES/NO
```

Para activar o no el envío de ficheros en el servidor por anónimos. En este caso, la autorización de escritura en un directorio es función de los derechos del directorio en el servidor (en particular si el usuario ftp tiene el derecho de escribir o no en un directorio):

```
anon_upload_enable=YES/NO
```

Puede prohibir la conexión a unos usuarios colocando sus nombres en `/etc/vsftpd.ftpusers`.

Puede añadir usuarios en `/etc/vsftpd.user_list` si `userlist_enable=YES`. En este caso, es el valor `deuserlist_deny` (YES/NO) el que determinará si el fichero contiene los usuarios prohibidos o autorizados.

Se puede crear en cada directorio del servidor un fichero `.message`. En este caso, su contenido se visualizará en el momento del acceso al directorio.

# Compartir Windows con Samba

## 1. Presentación

**Samba** es un conjunto de servidores que implementan los protocolos SMB/CIFS y NetBIOS/WINS para Unix. Su utilización más famosa es la de compartir recursos entre Windows y Unix, pero funciona perfectamente entre dos Unix. Una **compartición** se llama también **servicio**. Samba se compone de dos servicios:

- **smbd**: servidor SMB/CIFS.
  - Autenticación y autorización.
  - Ficheros e impresoras compartidos.
- **nmbd**: servidor de nombres NetBIOS.
  - Recorrido de los recursos.
  - Servidor WINS.

Un tercer servicio, **winbindd**, permite utilizar las cuentas de usuario de un dominio Microsoft. Las últimas versiones de Samba (3 y siguientes) permiten conectarse también a Active Directory.

Las funciones principales de Samba son:

- Autenticación de los usuarios.
- Ficheros e impresoras compartidos.
- Recorrido de los recursos compartidos de la red.
- Resolución de nombres (independiente de DNS) Nombre Netbios IP o viceversa.

## 2. Configuración

La configuración de Samba se encuentra en `/etc/samba/smb.conf`. Puede comprobar su sintaxis con la herramienta **testparm**.

El fichero `smb.conf` retoma la sintaxis de los ficheros de configuración de Windows de tipo **ini** con secciones delimitadas por corchetes **[ ]**.

Por defecto hay tres secciones:

- **[global]**: ajustes genéricos y globales del servidor, nombre, comentarios, método de autenticación, ajustes por defecto, etc.
- **[homes]**: compartición de los directorios personales de los usuarios.
- **[printers]**: compartición de las impresoras.

Los parámetros tienen la forma:

nombre = valor

Los comentarios comienzan por un punto y coma **,** o una almohadilla **#**.

```
[global]
workgroup = MYGROUP
```

```
netbios name = posteN  
security = share
```

- **workgroup**: nombre de grupo/dominio de trabajo;
- **netbios name**: nombre netbios de la máquina;
- **security**: método de autenticación (vea más adelante).

### 3. Ficheros compartidos

Cada partición debe disponer de su propia sección en smb.conf. Veamos cómo compartir el directorio /opt/compartición1 con el nombre de servicio **compartición1**:

```
[compartición1]  
comment = Directorio compartido 1  
path = /opt/compartición1  
browseable = yes  
public = no  
writable = yes  
printable = no  
group = compartición
```

Veamos una descripción de estos valores y de algunos otros posibles.

- **compartición1**: el nombre de la partición tal y como aparece en el «directorio de recursos compartidos».
- **comment**: el comentario tal como aparece al lado del nombre del recurso compartido.
- **path**: la ruta al recurso.
- **public**: el recurso está a disposición del invitado (guest).
- **browsable**: la compartición aparece en el «directorio de recursos compartidos».
- **writable**: el recurso está accesible en modo lectura y escritura.
- **printable**: la impresora está compartida.
- **group**: nombre del grupo por defecto para la conexión.
- **valid users**: nombre de los usuarios autorizados a acceder a este recurso.
- **read only**: el recurso está en modo de sólo lectura para todo el mundo.
- **guest ok**: ninguna contraseña es necesaria para acceder al recurso. En este caso, se utilizará la cuenta invitada por defecto.
- **guest only**: el recurso está accesible únicamente a los invitados.

Se autoriza el acceso a los recursos por defecto (ver la parte sobre los métodos de autenticación) en función de los derechos Unix. En el acceso al recurso, se requiere un nombre de usuario y una contraseña. Los derechos del directorio compartido y de su contenido determinan los derechos del usuario.

### 4. Impresoras compartidas

Además de la sección **[printers]**, puede añadir secciones para impresoras específicas. El

parámetro **printing** de la sección **[global]** permite modificar el subsistema de impresión basado por defecto en CUPS.

```
[Escritorio150]
comment = Laserjet 2100
printer = lj2100
valid users = juanito jorgito jaimito
path = /var/spool/lj2100
public = no
writable = no
printable = yes
browseable = yes
```

• **printer**: nombre de la impresora en Linux.

- **valid users**: nombre de los usuarios autorizados a acceder a esta compartición. Una @ delante del nombre indica un grupo de usuarios.
- **path**: ruta del spool de impresión.

## 5. Métodos de autenticación

Samba propone varios métodos de autenticación definidos en la sección **[global]**:

- **user**: método por defecto, el acceso al conjunto de las comparticiones de un servidor se efectúa mediante la validación de un nombre de usuario y de una contraseña únicos.
- **share**: método de validación de los identificadores compartición por compartición. En este caso, todos los accesos a las comparticiones, incluso públicas, necesitan identificadores.
- **domain**: utilización de un grupo de trabajo con autenticación.
- **ads**: utilización de Active Directory.

Otros tipos de autenticación son posibles como un servicio directorio LDAP.

```
[global]
workgroup = MYGROUP
netbios name = puestoN
security = share
```

## 6. Correspondencia de los nombres y contraseñas

Las contraseñas del protocolo SMB no tienen la misma forma que las contraseñas Unix/Linux. Hay que volver a crear las contraseñas para cada usuario que debe utilizar SMB con el comando **smbpasswd**. Los usuarios ya deben existir en Unix.

```
smbpasswd -a pepito
```

Los usuarios SMB están presentes en **/etc/samba/smbpasswd**. El comando **mksmbpasswd** puede realizar esto en batch:

```
cat /etc/passwd | mksmbpasswd > /etc/samba/smbpasswd
```

Puede establecer una tabla de correspondencia entre los nombres de usuarios de Windows y los de Unix en **/etc/samba/smbusers**.

```
# Unix_name = SMB_name1 SMB_name2 ...
root = administrator admin administrador
```

## 7. Clientes SAMBA

### a. En línea

Cualquier máquina en Microsoft Windows puede acceder a los recursos compartidos Samba. Los navegadores de los entornos de escritorio **KDE** y **GNOME** aceptan la navegación en las comparticiones gracias al protocolo **smb:/** en las URL. KDE propone el equivalente a un directorio de recursos compartidos.

La herramienta **smbclient** es un tipo de cliente FTP para el protocolo SMB. Las rutas de acceso a los recursos tienen la forma:

//máquina/compartición

Por ejemplo, para conectarse al servicio (aquí un recurso compartido) de una máquina:

```
smbclient //máquina/servicio -U login%passwd
```

Para listar los servicios propuestos por una máquina:

```
smbclient -L hostname -U login%passwd # lista de los recursos
```

### b. Montaje

Monte un sistema de ficheros SMB con el comando **smbmount**.

```
smbmount //máquina/compartición /mnt/mountpoint -o username=login
```

También se puede efectuar el montaje en `/etc/fstab`. Al igual que nfs, es un servicio especializado que montará y desmontará los recursos compartidos. Los últimos núcleos Linux sustituyeron **smbfs** por **cifs**:

```
//máquina/compartición /mnt/mountpoint cifs defaults,username=nobody 0 0
```

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

## **1. Preguntas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **Fundamentos TCP/IP**

**1** ¿Qué universidad integró IP en Unix?

- A - Harvard
- B - Berkeley
- C - Massachusetts
- D - Cambridge

**2** ¿Cuántos bits tiene una dirección IPv4?

**3** Entre las direcciones siguientes, ¿cuáles son válidas para un equipo en Internet?

- A - 171.27.43.10
- B - 172.18.25.128
- C - 34.352.28.17
- D - 192.169.194.69

**4** Una empresa utiliza un rango de dirección de tipo 10.x.y.z. ¿Cuál es la clase IP correspondiente?

**5** Esta misma empresa desea crear subredes de unas mil máquinas. ¿Cuál es la máscara de subred asociada?

**6** ¿Cómo se llama la pasarela hacia la cual se envían los paquetes cuando no hay otra posibilidad?

**7** En una empresa cuya subred está en 10/22 (255.255.252.0), la pasarela por defecto de cada subred es aquella en la que todos los últimos 10 bits están en 1 excepto el último, a 0. ¿Cuál es la IP de la pasarela de la subred 10.17.32.0?

### **Configuración IP**

**8** Configure manualmente eth0 con la IP 10.17.32.28/22 y active la interfaz.

**9** En Red Hat, ¿cuál es la ruta del fichero de configuración de la interfaz eth1?

**10** ¿Qué entrada del fichero de configuración de la tarjeta eth0 en Red Hat permite declarar la tarjeta como cliente DHCP?

- A - ONBOOT=DHCP
- B - MASTER=DCHP
- C - BOOTPROTO=DHCP
- D - IPADDR=DHCP

**11** Tanto en Red Hat como en Debian, ¿cómo parar y reiniciar la interfaz eth0?

**12** En Debian, ¿qué hacen las líneas «auto eth» e «iface eth2 inet dhcp» del fichero /etc/network/interfaces?

**13** ¿Cómo suprimir la ruta por defecto?

**14** ¿Qué puerto por defecto se utiliza para conectarse a un servidor FTP?

- A - 20
- B - 21
- C - 22
- D - 23

## **Herramientas básicas**

- 15** ¿Es aconsejable utilizar telnet?
- 16** ¿Cómo emitir 10 ping hacia todos los equipos de subred 10.17.32.28/22?
- 17** ¿A quién pertenece el dominio linuxes.org?
- 18** El tráfico de red sobre la tarjeta eth0 parece perturbado. Sospecha que hay un problema en la tarjeta o el cable. ¿Cómo comprobar el estado del tráfico?
- 19** ¿Cómo visualizar la tabla de encaminamiento de su máquina?
- A - route
  - B - netstat -n
  - C - netstat -r
  - D - netstat -rn
- 20** Un proceso bloquea el puerto en escucha 22. ¿Cómo puede usted aislar el nombre y el ID de este proceso?

## **Configuración general**

- 21** Dé la ruta del fichero que permite configurar el resolver.
- 22** Dispone de una lista de cuatro servidores DNS. ¿Puede colocar cuatro entradas «nameserver»?
- 23** Tiene cinco PC en su red personal, y no desea crear un servidor DNS. ¿Dónde colocará sus datos de resolución de nombre?
- 24** ¿Qué línea se debe modificar en /etc/nsswitch.conf para forzar la resolución DNS de los nombres en primer lugar?

- A - services: dns files
- B - networks: dns files
- C - hosts: dns files
- D - netgroup: dns files

- 25** ¿Cómo conocer el puerto de red asociado al servicio ntp?

- A - grep ^ntp /etc/services
- B - netstat -a | grep ntp
- C - ps -ef| grep ntp
- D - grep ntp /etc/protocols

- 26** ¿Qué ruta del fichero da la lista de los protocolos IP conocidos por el sistema?

## **Xinetd**

- 27** ¿Cuál es el papel de xinetd?
- 28** ¿Dónde se sitúan los ficheros de configuración de xinetd para cada servicio?
- 29** En Red Hat, ¿cuáles son los medios de activar un servicio controlado por xinetd?

- A - service xinetd start
- B - chkconfig service on
- C - disable = no dans /etc/xinetd.d/service
- D - # default: on dans /etc/xinetd.d/service

**30** A pesar de la activación del servicio telnet controlado por xinetd, éste parece no funcionar. ¿Qué tiene que hacer?

**31** xinetd ya no acepta conexiones y el período de espera es demasiado largo para las nuevas conexiones. ¿Cómo aceptar 60 conexiones por segundo y reducir el plazo a 10 segundos?

### **Módems y PPP**

**32** Su tarjeta módem PCI comprada por menos de 10 euros no parece funcionar en Linux. ¿Qué debe comprobar?

**33** ¿Cómo se llama el segundo periférico de serie propuesto mediante un adaptador USB?

- A - ttyACM1
- B - rfcomm1
- C - ttyS1
- D - ttyUSB1

**34** Su módem conectado en el primer puerto serie, con la norma V92, se conecta correctamente a 56 kbps a su proveedor, pero la comunicación con su PC sigue siendo muy lenta. ¿Cómo modificar la velocidad de este puerto en 115200 bps?

**35** ¿Con qué caracteres empiezan todos los comandos de control de los módems?

- A - CMD
- B - AT
- C - ATH
- D - ATDT

**36** ¿Es posible establecer una conexión PPP utilizando un móvil como módem?

**37** Dé la ruta de la ubicación de los ficheros de configuración PPP.

**38** ¿Por qué el programa pppd tiene el privilegio SUID activado?

**39** ¿Cómo forzar pppd a intentar cinco conexiones antes de desistir?

**40** ¿Qué nombre lleva el comando que establece el diálogo de identificación y autenticación con su PAI en el momento de la conexión?

- A - pppd
- B - telnet
- C - setserial
- D - chat

**41** ¿Qué nombre lleva la interfaz de red de una conexión ppp?

### **SSH**

**42** ¿Cómo prohibir una conexión directa como root mediante un cliente SSH?

**43** Con la ayuda del manual, ¿qué variable de configuración se debe modificar en sshd\_config para autorizar el encaminamiento del protocolo X?

**44** ¿Cómo ejecutar el comando remoto /tmp/runcmd.sh mediante ssh en la cuenta seb del servidor

slytest?

- A - ssh -l seb -c /tmp/runcmd.sh slytest
- B - ssh -u seb@slytest /tmp/runcmd.sh
- C - ssh seb@slytest /tmp/runcmd.sh
- D - ssh -l seb slytest -c /tmp/runcmd.sh

**45** Para establecer una conexión segura entre su cliente ssh en el servidor slytest sin introducir una contraseña, ¿qué debe usted hacer?

## **DHCP**

**46** ¿Qué significa DHCP?

**47** Dé la ruta del fichero de configuración del servidor dhcp.

**48** En la subred 10.17.32.0/22, ¿cuál debe ser el «rango» óptimo sabiendo que la pasarela es 10.17.35.254 y que las primeras 40 direcciones están reservadas?

**49** ¿Qué palabra clave permite, en asociación con la dirección MAC de una tarjeta Ethernet, asociar una IP fija a esta tarjeta?

**50** ¿Qué debe añadir a su configuración de servidor DHCP para que todos los clientes puedan recibir su nombre de anfitrión desde su servidor?

- A - opción host-name "nombre"; en el bloc host
- B - línea use-hostnames en la configuración general
- C - línea use-host-decl-name on; en la configuración subnet
- D - línea use-host-decl-name on; en la configuración general

**51** ¿Cómo activar manualmente la configuración DHCP de eth1 sin sobrescribir su fichero /etc/resolv.conf?

## **DNS**

**52** ¿Qué es la resolución de nombres?

**53** ¿Qué es la resolución inversa?

**54** ¿Cuál es la longitud máxima de un FQDN?

- A - 256 caracteres.
- B - 127, sin los puntos.
- C - 255 caracteres.
- D - 63 caracteres.

**55** ¿Cómo se llama el proceso que gestiona el servicio dns?

- A - bind
- B - named
- C - dnsd
- D - resolvd

**56** ¿Cuál es el papel del servidor esclavo?

**57** ¿Cuántos ficheros de configuración hacen falta como mínimo para crear un servidor DNS para un dominio o subdominio dado?

**58** ¿Cómo se debe llamar la zona de resolución inversa de la subred 10.17.32.0/22?

- A - 32.17.10.in-addr.arpa

- B - 10.17.32.0.in-addr.arpa
- C - el nombre que quiera
- D - 32.17.10.0.in-addr.arpa

**59** El servidor es de tipo master en 10.17.9.1, y el esclavo es el servidor 10.17.28.9. ¿Es posible?

**60** El nombre «www» es:

- A - el nombre de la red Web en Internet
- B - una simple convención para nombrar a un servidor Web
- C - un nombre de anfitrión como otro cualquiera
- D - un prefijo reservado para el servidor Web del dominio dado

**61** En el fichero de resolución, ¿cómo dar una IP al propio dominio?

**62** ¿Qué ocurre si usted efectúa una resolución inversa sobre un registro de tipo CNAME?

**63** Su zona de resolución empieza así «@ IN SOA dns1.slynet.org admin ( ... ». No funciona. ¿Cuál es su error?

**64** Su servidor DNS es dns1.pai.es. ¿Cómo obtener la lista de todos los anfitriones de este dominio (si el servidor lo autoriza)?

**65** ¿Cómo conocer el nombre de anfitrión asociado a la IP 10.17.35.54?

- A - dig -x 10.17.35.54
- B - host 10.17.35.54
- C - host -x 10.17.35.54
- D - nslookup 10.17.35.54

## **Correo electrónico**

**66** ¿Cuál es la cadena completa de envío, de traspaso y de recepción de correo?

- A - MTA -> MUA -> DMA -> MU
- B - MUA -> MTA -> MDA -> MUA
- C - MUA -> MTA -> ... -> MTA -> MDA -> MUA
- D - MUA -> MTA -> ... -> MTA -> MUA

**67** ¿Cuál es la ventaja de postfix en comparación con sendmail?

**68** Dé la ruta del fichero de configuración de postfix.

**69** ¿Qué línea de comandos permite especificar que el MTA es destinatario del correo del dominio slynet.org?

**70** ¿Qué fichero contiene la lista de los alias de usuarios para el correo electrónico?

**71** El usuario seb debe recibir el correo con destino a root. Del mismo modo; dispone de otro nombre, sebastien.rohaut. ¿Qué línea debe usted añadir en el fichero adecuado?

## **Apache**

**72** Su servidor Apache se niega a iniciarse (httpd start) después de editar su configuración. ¿Qué comando debe teclear?

**73** ¿Qué línea del fichero httpd.conf define la raíz del sitio Web?

- A - ServerRoot
- B - DocumentRoot

- C - Directory
- D - UserDir

- 74** Usted llama a su servidor Web por su dirección IP, pero contesta por su nombre. Ahora bien: su máquina no efectúa una resolución correcta y su navegador muestra una página de error. ¿Qué debe modificar?
- 75** Debe prohibir la visualización del contenido de un directorio desde un navegador. ¿Qué añade entre las etiquetas <Directory> correspondientes?

### **Ficheros compartidos**

- 76** ¿Cómo compartir /public en lectura y escritura para todo el mundo mediante NFS?
- 77** ¿Cómo visualizar las carpetas exportadas desde su máquina?
- 78** Liste los clientes que acceden a sus recursos compartidos.
- 79** Especifique la línea de /etc/fstab que permite montar el directorio /public de máquina1 en el /public de su propia máquina en nfs:
- 80** ¿Cuál es el nombre del principal servidor FTP más utilizado actualmente en Linux?
- A - proftpd
  - B - ftpd
  - C - wu-ftpd
  - D - vsftpd
- 81** ¿Está limitado Samba a los ficheros compartidos?
- 82** Dé el nombre de la herramienta que permite probar la configuración de Samba.
- 83** ¿Debe un cliente tener Samba instalado para acceder a recursos compartidos mediante Samba?
- 84** ¿Cómo simular una compartición de ficheros procedentes de Windows 9x?
- 85** ¿Qué líneas de smb.conf especifican la ruta a un recurso compartido?

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /85

Para este capítulo, su resultado mínimo debe ser de 64 respuestas acertadas (64/85).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- Fundamentos TCP/IP.
- Configuración IP.
- Herramientas básicas.
- Configuración general.
- Xinetd.
- Módems y PPP.
- SSH.
- DHCP.

- DNS.
- Correo electrónico.
- Apache.
- Ficheros compartidos.

### 3. Respuestas

#### **Fundamentos TCP/IP**

**1** ¿Qué universidad integró IP a Unix?

- A - Harvard
- B - Berkeley
- C - Massachusetts
- D - Cambridge

*B. Berkeley desarrolló IP con los fondos de la DARPA, bajo BSD.*

**2** ¿Cuántos bits tiene una dirección IPv4?

*32 bits.*

**3** Entre las direcciones siguientes, ¿cuáles son válidas para un equipo en Internet?

- A - 171.27.43.10
- B - 172.18.25.128
- C - 34.352.28.17
- D - 192.169.194.69

*A y D. La B es una dirección de clase B reservada a las redes locales, y la C es imposible ya que usa un valor que no es correcto (352).*

**4** Una empresa utiliza un rango de dirección de tipo 10.x.y.z. ¿Cuál es la clase IP correspondiente?

*Es la clase A.*

**5** Esta misma empresa desea crear subredes de unas mil máquinas. ¿Cuál es la máscara de subred asociada?

*1000+2=1002. La potencia de 2 aproximada es 2<sup>10</sup>, o sea, 1024. Ponga sus valores: diez ceros a la derecha, veintidós unos a la izquierda. La máscara es 10/22, o sea 255.255.252.0.*

**6** ¿Cómo se llama la pasarela hacia la cual se envían los paquetes cuando no hay otra posibilidad?

*Es la pasarela por defecto o Default Gateway.*

**7** En una empresa cuya subred está en 10/22 (255.255.252.0), la pasarela por defecto de cada subred es aquella en la que todos los últimos 10 bits están en 1 excepto el último, a 0. ¿Cuál es la IP de la pasarela de la subred 10.17.32.0?

*10.17.35.254.*

#### **Configuración IP**

**8** Configure manualmente eth0 con la IP 10.17.32.28/22 y active la interfaz.

```
ifconfig 10.17.32.28/22 up o también ifconfig 10.17.32.28 netmask 255.255.252.0 up.
```

**9** En Red Hat, ¿cuál es la ruta del fichero de configuración de la interfaz eth1?

```
/etc/sysconfig/network-scripts/ifcfg-eth1.
```

**10** ¿Qué entrada del fichero de configuración de la tarjeta eth0 en Red Hat permite declarar la tarjeta como cliente DHCP?

- A - ONBOOT=DHCP
- B - MASTER=DCHP
- C - BOOTPROTO=DHCP
- D - IPADDR=DHCP

*Es C, BOOTPROTO=DHCP.*

**11** Tanto en Red Hat como en Debian, ¿cómo parar y reiniciar la interfaz eth0?

```
ifdown eth0 && ifup eth0
```

**12** En Debian, ¿qué hacen las líneas «auto eth» e «iface eth2 inet dhcp» del fichero /etc/network/interfaces?

*La interfaz de red eth2 está activada en el boot como cliente DHCP.*

**13** ¿Cómo suprimir la ruta por defecto?

```
route del default gw
```

**14** ¿Qué puerto por defecto se utiliza para conectarse a un servidor FTP?

- A - 20
- B - 21
- C - 22
- D - 23

*B. Es el puerto 21.*

## **Herramientas básicas**

**15** ¿Es aconsejable utilizar telnet?

*Excepto para probar una conexión, no. El tráfico, incluidas las contraseñas, transitan de forma transparente.*

**16** ¿Cómo emitir 10 ping hacia todos los equipos de subred 10.17.32.28/22?

```
ping -c 10 -b 10.17.35.255
```

**17** ¿A quién pertenece el dominio linuxes.org?

*Teclee whois linuxes.org y mire el «Registrant Name».*

**18** El tráfico de red sobre la tarjeta eth0 parece perturbado. Sospecha que hay un problema en la tarjeta o el cable. ¿Cómo comprobar el estado del tráfico?

*Inserte netstat -i y mire los contadores de columnas RX-ERR, TX-ERR; en realidad, de todas las columnas menos RX-OK y TX-OK. Si el valor es diferente de 0, entonces hay un problema.*

**19** ¿Cómo visualizar la tabla de encaminamiento de su máquina?

- A - route
- B - netstat -n
- C - netstat -r
- D - netstat -rn

A, C y D. El comando route solo visualiza el mismo resultado que netstat -rn.

**20** Un proceso bloquea el puerto en escucha 22. ¿Cómo puede aislar el nombre y el ID de este proceso?

El puerto 22 es un puerto TCP netstart -A inet. El puerto está a la escucha y por lo tanto no debe restringir la visualización a las conexiones establecidas: -a . Para terminar, usted debe visualizar los procesos asociados: -p. La respuesta es netstat -A inet -a -p o, más sencillamente, netstat -apA inet.

### **Configuración general**

**21** Dé la ruta del fichero que permite configurar el resolver.

/etc/resolv.conf.

**22** Dispone de una lista de cuatro servidores DNS. ¿Puede colocar cuatro entradas «nameserver»?

No, está limitado a tres.

**23** Tiene cinco PC en su red personal, y no desea crear un servidor DNS. ¿Dónde colocará sus datos de resolución de nombres?

Añada las asociaciones IP Nombres en un fichero /etc/hosts y vuelva a copiar este fichero en cada uno de los puestos de la red.

**24** ¿Qué línea se debe modificar en /etc/nsswitch.conf para forzar la resolución DNS de los nombres en primer lugar?

- A - services: dns files
- B - networks: dns files
- C - hosts: dns files
- D - netgroup: dns files

C.

**25** ¿Cómo conocer el puerto de red asociado al servicio ntp?

- A - grep ^ntp /etc/services
- B - netstat -a | grep ntp
- C - ps -ef| grep ntp
- D - grep ntp /etc/protocols

A. El fichero /etc/services contiene la lista de las asociaciones servicios/puertos.

**26** ¿Qué ruta del fichero da la lista de los protocolos IP conocidos por el sistema?

/etc/protocols

## **Xinetd**

**27** ¿Cuál es el papel de xinetd?

*Es un superservicio que se encarga del acceso a otros servicios de redes y los controla, y que puede aplicar una configuración de acceso específica a cada uno de ellos.*

**28** ¿Dónde se sitúan los ficheros de configuración de xinetd para cada servicio?

*En el directorio /etc/xinetd.d.*

**29** En Red Hat, ¿cuáles son los medios de activar un servicio controlado por xinetd?

- A - service xinetd start
- B - chkconfig service on
- C - disable = no dans /etc/xinetd.d/service
- D - # default: on dans /etc/xinetd.d/service

*B y C. En realidad chkconfig modifica la línea disable en el fichero de configuración xinetd del servicio.*

**30** A pesar de la activación del servicio telnet controlado por xinetd, éste parece no funcionar. ¿Qué tiene que hacer?

*Un servicio controlado por xinetd no funciona si la configuración de xinetd no se actualiza de nuevo. Vuelva a ejecutar xinetd.*

**31** xinetd ya no acepta conexiones y el período de espera es demasiado largo para las nuevas conexiones. ¿Cómo aceptar 60 conexiones por segundo y reducir el plazo a 10 segundos?

*Modifique la línea «cps» en /etc/xinetd.conf de la manera siguiente: cps = 60 10.*

## **Módems y PPP**

**32** Su tarjeta módem PCI comprada por menos de 10 euros no parece funcionar en Linux. ¿Qué debe comprobar?

*Su módem es probablemente un winmodem. Compruebe su compatibilidad en los diversos sitios de ayuda en línea y en particular en el sitio <http://linmodems.org>.*

**33** ¿Cómo se llama el segundo periférico de serie propuesto mediante un adaptador USB?

- A - ttyACM1
- B - rfcomm1
- C - ttyS1
- D - ttyUSB1

*D. El nombre puede cambiar según la distribución y las reglas udev.*

**34** Su módem conectado en el primer puerto serie, con la norma V92, se conecta correctamente a 56 kbps a su proveedor, pero la comunicación con su PC sigue siendo muy lenta. ¿Cómo modificar la velocidad de este puerto en 115200 bps?

*Con el comando setserial: setserial /dev/ttyS0 baud base 115200*

**35** ¿Con qué caracteres empiezan todos los comandos de control de los módems?

- A - CMD
- B - AT
- C - ATH
- D - ATDT

*B: por AT, que significa «Cuidado» (Attention).*

**36** ¿Es posible establecer una conexión PPP utilizando un móvil como módem?

*Sí, si el teléfono móvil es reconocido como módem (en general USB) cuando está conectado por cable al PC o si dispone de servicios bluetooth asociados. ¡Cuidado con el coste de conexión!*

**37** Dé la ruta de la ubicación de los ficheros de configuración PPP.

*/etc/ppp.*

**38** ¿Por qué el programa pppd tiene el privilegio SUID activado?

*Porque debe disponer de un acceso total a los ficheros periféricos de los módems.*

**39** ¿Cómo forzar pppd a intentar cinco conexiones antes de desistir?

*Modifique el valor de maxfail en su script de conexión.*

**40** ¿Qué nombre lleva el comando que establece el diálogo de identificación y autenticación con su PAI en el momento de la conexión?

- A - pppd
- B - telnet
- C - setserial
- D - chat

*D: es el comando chat, en general iniciado mediante la línea connect del script de conexión y que puede coger sus valores de diálogo en otro script.*

**41** ¿Qué nombre lleva la interfaz de red de una conexión ppp?

*ppp0.*

## **SSH**

**42** ¿Cómo prohibir una conexión directa como root mediante un cliente SSH?

*Del lado servidor, modifique /etc/ssh/sshd\_config y pase el valor de PermitRootLogin a no.*

**43** Con la ayuda del manual, ¿qué variable de configuración se debe modificar en sshd\_config para autorizar el encaminamiento del protocolo X?

*El manual (`man sshd_config`) indica que se debe modificar el valor de X11Forwarding a Yes.*

**44** ¿Cómo ejecutar el comando remoto `/tmp/runcmd.sh` mediante ssh en la cuenta seb del servidor slytest?

- A - ssh -l seb -c /tmp/runcmd.sh slytest
- B - ssh -u seb@slytest /tmp/runcmd.sh
- C - ssh seb@slytest /tmp/runcmd.sh

- D - ssh -l seb slytest -c /tmp/runcmd.sh

C.

**45** Para establecer una conexión segura entre su cliente ssh en el servidor slytest sin introducir una contraseña, ¿qué debe usted hacer?

*Debe copiar el contenido de la clave pública de la cuenta de su cliente (su cuenta en su máquina) en el fichero authorized\_keys2 de la cuenta a la cual usted accede en la máquina remota.*

## **DHCP**

**46** ¿Qué significa DHCP?

*Dynamic Host Configuration Protocol.*

**47** Dé la ruta del fichero de configuración del servidor dhcp.

*/etc/dhcpd.conf.*

**48** En la subred 10.17.32.0/22, ¿cuál debe ser el «rango» óptimo sabiendo que la pasarela es 10.17.35.254 y que las primeras 40 direcciones están reservadas?

*range 10.17.32.41 10.17.35.253*

**49** ¿Qué palabra clave permite, en asociación con la dirección MAC de una tarjeta Ethernet, asociar una IP fija a esta tarjeta?

*Es fixed-address, que debe colocarse en un bloque anfitrión.*

**50** ¿Qué debe añadir a su configuración de servidor DHCP para que todos los clientes puedan recibir su nombre de anfitrión desde su servidor?

- A - opción host-name "nombre"; en el bloc host
- B - línea use-hostnames en la configuración general
- C - línea use-host-decl-name on; en la configuración subnet
- D - línea use-host-decl-name on; en la configuración general

*D. La A limita a un anfitrión. La C limita a los clientes de esta única subred.*

**51** ¿Cómo activar manualmente la configuración DHCP de eth1 sin sobreescribir su fichero /etc/resolv.conf?

*dhpcd -R eth1 &*

## **DNS**

**52** ¿Qué es la resolución de nombres?

*El hecho de convertir un nombre de anfitrión en dirección IP.*

**53** ¿Qué es la resolución inversa?

*El hecho de convertir una dirección IP en nombre de anfitrión.*

**54** ¿Cuál es la longitud máxima de un FQDN?

- A - 256 caracteres.
- B - 127, sin los puntos.

- C - 255 caracteres.
- D - 63 caracteres.

C. 255 caracteres.

**55** ¿Cómo se llama el proceso que gestiona el servicio dns?

- A - bind
- B - named
- C - dnsd
- D - resolv

B. El producto se llama bind, pero el servicio se llama named.

**56** ¿Cuál es el papel del servidor esclavo?

Sirve de servidor secundario que contiene una copia en caché de los datos del maestro por si éste no contesta. Se debe declarar en /etc/resolv.conf, por supuesto.

**57** ¿Cuántos ficheros de configuración hacen falta como mínimo para crear un servidor DNS para un dominio o subdominio dado?

3: el fichero named.conf, el fichero de resolución de nombres y el fichero de resolución inversa.

**58** ¿Cómo se debe llamar la zona de resolución inversa de la subred 10.17.32.0/22?

- A - 32.17.10.in-addr.arpa
- B - 10.17.32.0.in-addr.arpa
- C - el nombre que quiera
- D - 32.17.10.0.in-addr.arpa

B. Debe invertir el orden para que named identifique más rápidamente la subred asociada.

**59** El servidor es de tipo master en 10.17.9.1, y el esclavo es el servidor 10.17.28.9. ¿Es posible?

Sí, ipor supuesto! Los servidores DNS se pueden situar en cualquier parte debajo de la red y gestionar todo los dominios deseados, a condición de que el master y el esclavo puedan comunicarse (encaminamiento).

**60** El nombre «www» es:

- A - el nombre de la red Web en Internet
- B - una simple convención para nombrar a un servidor Web
- C - un nombre de anfitrión como otro cualquiera
- D - un prefijo reservado para el servidor Web del dominio dado

B y C. Por convención, designa el servidor o los servidores Web de un dominio, pero se trata de una convención y nada más.

**61** En el fichero de resolución, ¿cómo dar una IP al propio dominio?

@ IN A IP

**62** ¿Qué ocurre si usted efectúa una resolución inversa sobre un registro de tipo CNAME?

*El servidor va a recuperar el «verdadero» nombre asociado al CNAME y va a efectuar la resolución inversa encima. Usted obtendrá la IP correcta.*

**63** Su zona de resolución empieza así «@ IN SOA dns1.slynet.org admin ( ...». No funciona. ¿Cuál es su error?

*iSe ha olvidado el punto después de dns1.slynet.org! Named va a concatenar: dns1.slynet.org.slynet.org.*

**64** Su servidor DNS es dns1.pai.es. ¿Cómo obtener la lista de todos los anfitriones de este dominio (si el servidor lo autoriza)?

*Con dig: dig @dns1.pai.es pai.es axes*

**65** ¿Cómo conocer el nombre de anfitrión asociado a la IP 10.17.35.54?

- A - dig -x 10.17.35.54
- B - host 10.17.35.54
- C - host -x 10.17.35.54
- D - nslookup 10.17.35.54

*A y B. nslookup no se debería utilizar ya.*

### **Correo electrónico**

**66** ¿Cuál es la cadena completa de envío, de traspaso y de recepción de correo?

- A - MTA -> MUA -> DMA -> MU
- B - MUA -> MTA -> MDA -> MUA
- C - MUA -> MTA -> ... -> MTA -> MDA -> MUA
- D - MUA -> MTA -> ... -> MTA -> MUA

*C. Puede haber un paso por varios MTA antes de llegar al MDA. La respuesta B puede ser correcta para un envío local (mismo dominio).*

**67** ¿Cuál es la ventaja de postfix en comparación con sendmail?

*Es más ligero y simple de configurar.*

**68** Dé la ruta del fichero de configuración de postfix.

*/etc/postfix/main.cf.*

**69** ¿Qué línea de comandos permite especificar que el MTA es destinatario del correo del dominio slynet.org?

*postconf -e "mydestination = slynet.org"*

**70** ¿Qué fichero contiene la lista de los alias de usuarios para el correo electrónico?

*/etc/aliases.*

# Prácticas

## 1. Configuración TCP/IP de Linux

Objetivo: configurar una agrupación de tarjetas de redes en una distribución de tipo Red Hat.

1. Es posible crear agrupaciones de tarjetas de redes con Linux. En caso de que falle una, otra puede tomar el relevo, o incluso n tarjetas pueden funcionar de manera simultánea. Las tarjetas se llaman bond0, bond1, etc.  
Cree un fichero ifcfg-bond0 en /etc/sysconfig/network-scripts. Complételo como cualquier otro fichero de configuración de red:

```
DEVICE=bond0
ONBOOT=yes
NETMASK=255.255.255.0
IPADDR=192.168.1.25
BOOTPROTO=static
```

2. Crear agrupaciones de tarjetas (bonding) necesita la carga del módulo «bonding». Los parámetros de este módulo determinan el modo de funcionamiento de la agrupación. La opción «modo» del módulo coge el valor 1 para un backup: si la primera tarjeta deja de funcionar, la segunda toma el relevo, y viceversa. Modifique el fichero de configuración /etc/modprobe.conf:

```
alias bond0 bonding
options bond0 mode=1
```

3. Las dos tarjetas eth0 y eth1 formarán parte de la agrupación. Modifique los ficheros de configuración de cada una de las tarjetas en consecuencia. Por ejemplo, para la tarjeta eth0, el fichero ifcfg-eth0:

```
DEVICE=eth2
HWADDR=00:1B:2C:3D:4E:5F
ONBOOT=yes
TYPE=Ethernet
MASTER=bond0
SLAVE=yes
```

4. Active la agrupación con el comando ifup:

```
# ifup bond0
```

5. Verifique el estado del bonding:

```
# cat /proc/net/bonding/bond0
Bonding Mode: fault-tolerance (active-backup)
Primary Slave: None
Currently Active Slave: eth0
...
Slave Interface: eth0
MII Status: up
Link Failure Count: 0
Permanent HW addr: xx:xx:xx:xx:xx:xx

Slave Interface: eth1
MII Status: up
```

```
Link Failure Count: 0  
Permanent HW addr: xx:xx:xx:xx:xx:xx
```

## 2. Algunos comandos de red

Objetivo: manejar los comandos ping y traceroute para determinar el funcionamiento de la red.

1. Emite un solo ping hacia la dirección que elija; por ejemplo, linuxes.org, y verifique el retorno del comando.

```
# ping -c 1 linuxes.org  
PING linuxes.org (88.191.250.104) 56(84) bytes of data.  
64 bytes from linuxes4 (88.191.250.104): icmp_seq=1 ttl=59  
time=6.43 ms  
  
--- linuxes.org ping statistics ---  
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 6.439/6.439/6.439/0.000 ms  
# echo $?  
0
```

2. El código de retorno 0 indica que la máquina remota ha contestado. Escriba una línea de comandos que permita probar si un servidor contesta y que, en caso contrario, permita salir del programa con un mensaje de error:

```
# ping -c 1 linuxes.org >/dev/null 2>&1 || { echo "El servidor no responde"; exit 1; }
```

3. Dos servidores de datos están en sitios diferentes. Un servidor cualquiera, esté donde esté, quiere determinar el servidor más cercano para limitar el tráfico de red y el número de pasarelas que debe pasar. Para ello, va a utilizar traceroute y las opciones -q 1 (una petición por salto) y -n (visualización de los «hops» en formato simple).

```
# traceroute -q 1 -n server1  
1  10.x.y.247  0.619 ms  
2  10.x.y.2  0.956 ms  
3  10.x.y.8  1.028 ms  
4  172.x.y.101  1.979 ms
```

4. Para determinar el trayecto más corto, basta con contar el número de saltos:

```
S1=$(traceroute -q 1 -n $SERVER1 2>/dev/null | wc -l)  
S2=$(traceroute -q 1 -n $SERVER2 2>/dev/null | wc -l)
```

5. Compare los dos valores y determine el servidor más cercano que hay que configurar en la variable SERVER; es el servidor en esta variable el que habrá que utilizar para optimizar los accesos.

```
[ $SERVER1 -ge $SERVER2 ] && SERVER=$SERVER2 || SERVER=$SERVER1
```

## 3. El resolver

Objetivo: gestionar la resolución de nombre.

1. Su máquina forma parte de una gran red que dispone de dos servidores de nombres DNS y varios dominios y subdominios declarados. Perteneces usted al dominio subnet1.slynet.org. Las IP de los servidores DNS son 10.17.23.127 y 10.17.23.247. Construya el fichero /etc/resolv.conf:

```
domain subnet1.slynet.org
nameserver 10.17.23.127
nameserver 10.17.23.247
```

2. Su máquina se llama s14p23 (sala 14, puesto 23). Debe poder acceder mediante su nombre corto (hostname) a todas las máquinas, incluyendo las de los otros subdominios. Por ejemplo, s19p01 está en el subdominio subnet2.slynet.org. Modifique /etc/resolv.conf en consecuencia:

```
domain subnet1.slynet.org
search subnet1.slynet.org subnet2.slynet.org
nameserver 10.17.23.127
nameserver 10.17.23.247
```

3. Algunos servicios recuperan el nombre de su máquina mediante una resolución local que pasa por /etc/hosts. Su IP es 10.17.35.168. Ahora bien: en Linux se suele asociar el nombre de anfitrión al localhost 127.0.0.1. Rectifique esto en /etc/hosts:

```
127.0.0.1      localhost
10.17.35.168    s14p23 subnet1.slynet.org
```

4. La resolución local pasa primero por /etc/hosts y luego por el DNS. Pero el DNS suele estar más actualizado que /etc/hosts. Modifique el fichero /etc/nsswitch en consecuencia: modifique el orden de la línea hosts:

```
hosts: dns files
```

## 4. Servicios de red

Objetivo: controlar varios servicios de redes.

1. Quiere saber a qué servicio está asociado el puerto 137. Busque la información en el fichero /etc/servicios. Se utiliza este puerto para la resolución de nombres Netbios (Microsoft, Samba).

```
# grep "137/" /etc/servicios
netbios-ns 137/tcp      #NETBIOS Name Service
netbios-ns 137/udp
```

2. Para todas las distribuciones, sin distinción, desactive el servicio xinetd telnet si está presente. Diríjase a /etc/xinetd.d, abra el fichero telnet y modifique la línea siguiente:

```
disable = yes
```

Vuelva a iniciar xinetd para leer la configuración de nuevo.

3. El servicio sshd autoriza la conexión de root y quiere evitarlo. Modifique la configuración de sshd en consecuencia. El fichero de configuración es /etc/ssh/sshd\_config. Modifique el parámetro siguiente:

```
PermitRootLogin no
```

O comente la línea simplemente y vuelva a ejecutar sshd.

4. Su máquina dispone de varios programas arrancados que establecen conexiones en red. Una de ellas -no sabe cuál- está conectada al puerto remoto 1863. ¿Cómo encontrar el proceso asociado?  
Un método consiste en utilizar netstat para listar las conexiones en curso y los procesos asociados:

```
# netstat -A inet -p -n | grep 1863
tcp      0      0 192.x.y.z    207.46.27.19:1863      ESTABLISHED
10505/kopete
```

Otra solución, que no se ha considerado en el capítulo, es la utilización de lsof con el parámetro `-i`. Ya que todo es fichero, las conexiones de red también:

```
# lsof -i :1863
COMMAND   PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
kopete  10505 seb    15u  IPv4 132412  0t0    TCP slyserver:33097
...
```

5. El funcionamiento por defecto del cliente DHCP consiste en sobreescribir el fichero `/etc/resolv.conf` con cada renovación de dirección. El parámetro `-R` lo evita. Según la distribución, usted puede impedir esto.  
Por ejemplo en openSUSE, el fichero `/etc/sysconfig/network/dhcp` acepta un parámetro `DHCLIENT MODIFY_RESOLV_CONF="yes" o "no"` para resolver por defecto este caso.  
Busque en la ayuda de su distribución cómo modificar este tipo de parámetros.
6. La «artimaña» siguiente le explica cómo mandar un correo electrónico desde una conexión telnet hacia un servidor SMTP. Abra una conexión telnet y siga las instrucciones siguientes:

```
# telnet smtp.dominio.net 25
HELO hola
MAIL FROM:joe.dalton@lucky.net
RCPT TO: sheriff@daisytown.net
DATA
subject:Nunca me tendrás

Sheriff, nunca cogerás a los Dalton.

.
QUIT
```

## 5. Ficheros compartidos

Objetivo: compartir ficheros y directorios mediante NFS y Samba. Se deben instalar e iniciar los servicios NFS y Samba.

1. En su máquina 192.168.1.25 cree un directorio `/DATOS` que va a compartir tanto mediante NFS como con Samba. Para NFS, modifique el fichero `/etc/exports` con el fin de exportar su contenido para todos los miembros de la subred 192.168.1.0/24 en modo de sólo lectura.

```
# mkdir /DATOS
# vi /etc/exports
```

Añada la línea siguiente:

```
/DATOS 192.168.1.0/255.255.255.0(ro)
```

2. Exporte su recurso. Verifique que el recurso compartido esté correctamente activo.  
Ejecute el comando `exportfs`:

```
# exportfs -av
```

Verifique si el recurso compartido está presente:

```
# showmount -e localhost
```

3. En otra máquina de la subred, monte esta compartición NFS en el directorio /DATOS local y convierta esta modificación en permanente:

```
# mount -t nfs 192.168.1.25:/DATOS /DATOS
```

Añada una línea en /etc/fstab de la manera siguiente:

```
192.168.1.25:/DATOS/DATOSnfsdefaults00
```

4. Exporte /DATOS con Samba, en modo de sólo lectura, únicamente para el usuario joe, con el nombre DATOS. El recurso compartido está oculto. Modifique el fichero /etc/smb/smb.conf así:

```
[DATOS]
comment = Carpeta compartida DATOS
path = /DATOS
browseable = no
public = no
writable = no
printable = no
valid users = joe
```

Pruebe la sintaxis:

```
# testparm
```

Y vuelva a ejecutar Samba.

5. Cree el usuario Samba joe asociado:

```
# smbpasswd -a joe
```

6. Pruebe el recurso compartido con smbclient:

```
# smbclient //192.168.1.25/DATOS -U joe
```

# TCP/IP

## 1. Fundamentos

El origen de **TCP/IP** se sitúa en las investigaciones de la **DARPA** (*Defense Advanced Research Project Agency*), que empezaron en 1970 y desembocaron en **ARPANET**. En realidad, la DARPA financió a la universidad de Berkeley, que integró los protocolos básicos de TCP/IP dentro de su sistema **UNIX BSD 4**.

TCP/IP se popularizó gracias a su interfaz genérica de programación de intercambios de datos entre las máquinas de una red, las primitivas **sockets**, y la integración de protocolos de aplicación. El **IAB** (*Internet Activities Board*) supervisa los protocolos de TCP/IP y a otros dos organismos:

- La **IRTF** (*Internet Research Task Force*), responsable del desarrollo de los protocolos.
- La **IETF** (*Internet Engineering Task Force*), responsable de la red Internet.

El **NIC** (*Network Information Center*) y en España el **ESNIC** distribuyen las direcciones red. Se describe el conjunto de los protocolos de TCP/IP en los documentos **RFC** (*Request For Comments*) (vea el RFC 793).

- La capa inferior es **IP** (*Internet Protocol*).
- La capa de transporte es **TCP** (*Transmission Control Protocol*) o **UDP** (*User Datagram Protocol*).
- Las capas superiores son las capas de los protocolos de aplicación; por ejemplo:
  - **NFS** (*Network File System*): compartición de los ficheros a distancia.
  - **DNS** (*Domain Name System*): asociación anfitrión<->IP.
  - **FTP** (*File Transfer Protocol*): transferencia de ficheros.
  - **TELNET**: emulación de un terminal de tipo texto...

La versión del protocolo IP representado es la V4. El futuro, ya presente, es el protocolo IPV6. Compatible con IPV4, propone un sistema de direcciones en 128 bits (16 bytes) que permite extender las capacidades de la red, en particular en cuestión de tamaño y direccionamiento.

## 2. Direccionamiento

### a. Clases

Es importante saber antes de la instalación a qué tipo de red se debe integrar el nuevo servidor, TCP/IP por supuesto, pero hay que reservarle una dirección IP, un hostname (nombre de máquina de red), conocer las diversas pasarelas, el nombre de dominio, la clase utilizada y la máscara de subred o netmask.

Hagamos un breve repaso de las clases de IP. Se define una dirección IP en 32 bits y se representa con cuatro nombres separados por puntos: **n1.n2.n3.n4**. Esta dirección está constituida por dos partes que definen la dirección de red y el anfitrión en la red, respectivamente.

Según los casos, se distinguen cuatro o cinco clases de direcciones: A, B, C, D y E, pero sólo las tres primeras nos interesan.

*Leyenda: N y h son bits; N es el identificador de la red h, que es el identificador de la máquina.*

**Clase A:** 0NNNNNNN hhhhhh hhhhhhh hhhhhhh o sea 1.x.x.x a 126.x.x.x.

$n_1$  está comprendido entre 1 y 126.  
16777214 anfitriones, 127 redes.

**Clase B:** 10NNNNNN NNNNNNNN hhhhhh o sea de 128.0.x.x a 191.255.x.x.  
 $n_1$  está comprendido entre 128 y 191.  
65534 anfitriones, 16382 redes.

**Clase C:** 110NNNNN NNNNNNNN NNNNNNNN hhhhhh o sea de 192.0.0.x a 223.255.255.x.  
 $n_1$  está comprendido entre 192 y 223.  
254 anfitriones, 2097150 redes.

**Clase D:** Empieza por 1110, para la multidifusión IP.

**Clase E:** Empieza por 1111 para experimentación.

Existen direcciones de anfitriones que no pueden ser utilizadas. Por ejemplo, en la clase C sólo se pueden tener 254 anfitriones, mientras que el identificador de la máquina está codificado en 8 bits (por lo tanto, 256 valores). La dirección 0 representa la dirección de la red, y la dirección 255, la del **broadcast** (multidifusión).

Observe que las direcciones siguientes no se deben enrutar en Internet y se reservan a redes locales.

- 10.0.0.0 - 10.255.255.255 (10/8)
- 172.16.0.0 - 172.31.255.255 (172.16/12)
- 192.168.0.0 - 192.168.255.255 (192.168/16)

La dirección 127.0.0.1 es la dirección de loopback o bucle: representa a la propia máquina, así como la subred 127.0.0.0/8.

## b. Subredes

Además, es posible dividir estas redes en subredes usando máscaras que permiten una división más pequeña de las direcciones. Un **netmask** es una máscara binaria que permite separar de manera inmediata la dirección de la red y de la subred, de la dirección del anfitrión en la dirección IP global. Las máscaras predefinidas son:

- **Clase A:** 255.0.0.0
- **Clase B:** 255.255.0.0
- **Clase C:** 255.255.255.0

Para comunicarse directamente entre ellos, los anfitriones deben pertenecer a una misma red o subred. Calcular una subred es bastante sencillo. Veamos un ejemplo para una red de clase C.

- Red: 192.168.1.0
- Dirección de red: 192.168.1.255
- Máscara de red: 255.255.255.0

*Calcular una máscara de subred:*

- Para calcular la máscara de subred, primero debe determinar cuántas máquinas quiere integrar en ella. Una red de clase C permite integrar 254 máquinas (0 y 255 están reservados). Desea crear redes que contienen 60 máquinas. Añada 2 a este valor para las direcciones reservadas (dirección de la subred y dirección de broadcast), lo que da **62**.
- Una vez determinado el número de máquinas, encuentre la potencia de dos, exacta o superior al número

encontrado. 2 elevado a 6 da **64**.

- Escriba la máscara en binario, coloque todos los bits de la máscara de red de clase C a 1, y coloque a 0 los 6 primeros bits de la máscara correspondiente a la parte máquina:  
**11111111 11111111 11111111 11000000**
- Convierta esta máscara en decimal: **255.255.255.192**, y calcule el conjunto de las subredes posibles. Como está en una red de clase C, todavía puede hacer variar los dos últimos bits de la parte máquina:
  - 00xxxxxx: 255.255.255.0
  - 01xxxxxx: 255.255.255.64
  - 10xxxxxx: 255.255.255.128
  - 11xxxxxx: 255.255.255.192
- Al final, obtiene cuatro subredes de 62 máquinas, o sea, 248 máquinas. En efecto: obtiene 256 si añade las cuatro direcciones de broadcast y las cuatro direcciones de red.

### c. Encaminamiento

La máscara de red permite determinar si una máquina destinataria está en la misma red que usted o no. Hay que indicar la ruta que deben tomar los paquetes IP para alcanzar su destino. Si su máquina es un puesto cliente que dispone de una sola tarjeta de red y esta red comporta un único enrutador (caso clásico de una conexión hacia Internet), entonces debe crear dos rutas. La primera es la que indica qué tarjeta de red deben emplear los paquetes para acceder al resto de la red (a la subred); la segunda, qué ruta deben utilizar los paquetes para salir de la red. En general, se habla de ruta por defecto cuando un solo enrutador está presente.

- Hacia red1 -> utilizar interfaz de red izquierda.
- Hacia red2 -> utilizar interfaz de red derecha.
- Hacia otras -> utilizar interfaz de red derecha hacia enrutador1.

Ejemplo: Red1 de clase C 192.168.1.0 en eth0, Red2 de clase B 172.16.0.0 en eth1, dirección de enrutador 192.168.1.254.

Red	Máscara	Interfaz	Pasarela
192.168.1.0	255.255.255.0	eth0	eth0
172.16.0.0	255.255.0.0	eth1	eth1
0.0.0.0	0.0.0.0	eth0	192.168.1.254

Todos los paquetes de red hacia 192.168.1.0 transitarán por eth0. Todos los paquetes cuyo destino sea 172.16.0.0 transitarán por eth1. Por defecto, todos los otros paquetes para las redes no especificadas transitarán por eth0 y serán tratados por la pasarela 192.168.1.254, que enrutará los paquetes.

### d. IPv6

#### Saturación de IPv4

Las últimas direcciones IPv4 públicas llegaron a su saturación el 3 de febrero de 2011, aunque no se usen todas ellas debido al modo de repartición por clases y subredes. Por ejemplo, las únicas subredes de clase A representan cerca de dos mil millones de direcciones IP, probablemente sin usarse todas o en manos de organismos que no usan todas. Incluso sucede lo mismo con las subredes de clase B, con mil millones de direcciones.

Un gran número de direcciones IP se reservan para usos concretos, como por ejemplo las IP multicast, que no pueden asignarse a un equipo.

Una mala gestión inicial, la multiplicación de los terminales, especialmente de terminales móviles de tipología diversa y el gran crecimiento en número de conexiones a Internet han incrementado la demanda de forma espectacular, lo que nos ha llevado a su extinción.

Se han aplicado varios métodos para intentar retrasar la fecha de saturación, especialmente haciendo desaparecer el concepto de clases en las IP públicas o el uso de NAT, que permite mediante la traducción de direcciones ofrecer servicios accesibles al público alojados en máquinas pertenecientes a una subred privada. Es así como usted puede instalar servidores web u otros servicios en sus propios PC de casa, que tienen una dirección IP de una red privada, pero son accesibles desde el exterior mediante la IP pública y las funciones NAT de su router de Internet.

Las direcciones IPv4 no usadas que todavía están en manos de sus propietarios originales ahora son objeto de un negocio lucrativo.

Todo esto no significa la inmediata desaparición de IPv4: estas direcciones IP por supuesto se seguirán usando durante bastantes años, el tiempo en el que todas las IP no utilizadas dejen de serlo, lo que se debería producir entre 2011 y 2015. Las direcciones IPv4 públicas están llamadas a desaparecer o a cohabitar con otro mecanismo.

## **Direccionamiento IPv6**

Las direcciones IPv6 reemplazarán a las direcciones IPv4. Se codifican con 128 bits y hay tantas disponibles que cada ser humano podría disponer de un trillón de direcciones sin llegar a saturarse. De este modo, NAT se convierte en una tecnología inútil y el enrutamiento se vuelve más sencillo.

La dirección se descompone en 8 grupos de dos bytes escritos en hexadecimal y separados por ":", es decir 39 caracteres:

2001:0e36:2ed9:d4f0:021b:0000:0000:f81d

Pueden omitirse de uno a tres ceros no significativos:

2001:e36:2ed9:d4f0:21b:0:0:f81d

Uno o varios bloques de 16 bytes (2 bytes) nulos pueden omitirse conservando los ":" de cada lado. En el ejemplo usado anteriormente, sólo se puede aplicar una solución.

2001:e36:2ed9:df40:21b:0:0:f81d

El prefijo predeterminado para las direcciones es /64. Lo que significa que su proveedor de Internet le proporcionará una dirección con el formato 2a01:e36:2ed9:d4f0::/64, es decir de forma desarrollada:

2001:0e36:2ed9:d4f0:0000:0000:0000:0000/64

Esta subred es suya y se podrá direccionar automáticamente por su proveedor de Internet. No se puede (todavía) subdividirla en otras subredes. Podrá crear tantas direcciones IPv6 como desee en esta subred, alrededor de  $2^{64}$  huéspedes: 18.446.744.073.709.551.616 direcciones IP. Estas IP serán directamente accesibles desde el exterior, sin NAT, pero por supuesto podrá proteger sus equipos con un firewall.

## **Uso en Linux**

Este libro se basa en la configuración de las interfaces y de los servicios para un uso en IPv4. Sin embargo, el principio es el mismo que en IPv6. Linux es completamente compatible con IPv6, de forma nativa. Los comandos ifconfig, route, ping, etc. aceptan IPv6, así como los archivos de configuración de las interfaces propias de cada distribución. A continuación se muestran algunos ejemplos adaptados.

### 3. Configuración

#### a. El caso de las distribuciones de tipo Red Hat/Fedora

Red Hat propone herramientas para configurar la red básica sin pasar por el manejo de los ficheros de configuración. El programa en modo de texto **netconfig** permite instalar la configuración TCP/IP básica (IP estática o dinámica, enrutador, nombre de anfitrión, servidor de nombres).

```
# netconfig --device eth0
```

El comando gráfico **system-config-network** inicia una interfaz más completa.

Las distribuciones Mandriva y openSUSE retoman el mismo principio que el que se expone a continuación, pero la sintaxis y la posición de los ficheros pueden variar. Vea la documentación de su distribución para más detalles.

#### Interfaces de red

La configuración básica de una interfaz de red se hace con la ayuda del comando **ifconfig**. Sin embargo, la mayoría de las distribuciones utilizan scripts de administración y ficheros de configuración que simplifican mucho las cosas, ya que configuran al mismo tiempo la interfaz de red y las rutas.

##### Configuración de eth0 para la dirección de clase C 192.168.1.2

```
ifconfig eth0 inet 192.168.1.2 netmask 255.255.255.0  
ifconfig eth0 192.168.1.2
```

##### Activación de la interfaz de red eth0:

```
ifconfig eth0 up
```

##### Parada de la interfaz de red eth0:

```
ifconfig eth0 down
```

##### Visualización de información de eth0:

```
# ifconfig eth0  
eth0      Vínculo encap:Ethernet  HWaddr 00:XX:XX:XX:XX:XX  
          inet adr:192.168.1.60  Bcast:192.168.1.255  Máscara:255.255.255.0  
          adr inet6: fe80::21b:fcff:fea9:f81d/64 Scope:Vínculo  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:16522 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:13631 errors:0 dropped:0 overruns:0 carrier:2  
          collisions:0 lg file transmission:1000  
          RX bytes:17732221 (16.9 MB)  TX bytes:1648879 (1.5 MB)
```

Observe que la dirección IPv6 aparece en el resultado en la línea `inet6`.

##### Configuración de una dirección IPv6

```
# ifconfig eth0 inet6 add fe80::21b:fcff:fea9:f81d/64
```

##### Visualización de todas las interfaces de red activadas:

```
ifconfig
```

### Visualización de todas las interfaces de red, activadas o no:

```
ifconfig -a
```

Lo mejor sigue siendo la utilización de los scripts **ifup** e **ifdown**. Se basan en los ficheros presentes en /etc/sysconfig/network-scripts/. Estos ficheros de configuración de interfaz se llaman ifcfg-xxx, donde xxx es el nombre de la interfaz de red, como eth0:

```
DEVICE=eth0
IPADDR=192.168.1.2
NETMASK=255.255.255.0
NETWORK=192.168.1.0
BROADCAST=192.168.1.255
ONBOOT=yes
BOOTPROTO=static
```

Los parámetros hablan por sí mismos. Los valores **NETWORK** y **BROADCAST** son opcionales si se notifican **IPADDR** y **NETMASK** (en este caso, el cálculo es automático) o si se utiliza **DHCP**. **BOOTPROTO** indica cómo montar la interfaz, o bien **static**, o bien **dhcp**. Se puede utilizar el valor **bootp**. En el caso de DHCP, el fichero se puede parecer a esto:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
```

En el caso de una configuración estática, **IPADDR** y **NETMASK** son obligatorios:

```
DEVICE=eth0
IPADDR=192.168.1.2
NETMASK=255.255.255.0
ONBOOT=yes
BOOTPROTO=static
```

**ONBOOT** determina si se debe activar automáticamente la interfaz en el arranque de la máquina.

Para IPv6, usted debe utilizar las siguientes variables:

```
IPV6INIT=yes
IPV6ADDR=<IPv6-IP-Address>
IPV6_DEFAULTGW=<IPv6-IP-Gateway-Address>
```

**IPV6INIT** indica si IPv6 está activo o no en esta interfaz. Las dos variables siguientes indican la dirección IP y su puerta de enlace.

Una vez notificado correctamente el fichero, se utilizan los comandos **ifup/ifdown**:

### Activación de la interfaz eth0:

```
ifup eth0
```

### Parada de la interfaz eth0:

```
ifdown eth0
```

### **Parámetros generales**

El fichero /etc/sysconfig/network contiene los parámetros generales de la red.

```
NETWORKING=yes
HOSTNAME=puestol.mired.org # nombre completo
GATEWAY=0.0.0.0 # pasarela por defecto
NISDOMAIN= # nombre del dominio NIS
NETWORKING_IPV6=yes
```

- **NETWORKING**: activación o no de la red.
- **HOSTNAME**: nombre de dominio completo FQDN.
- **GATEWAY**: dirección IP de la pasarela.
- **GATEWAYDEV**: interfaz de red que permite acceder a la pasarela.
- **NISDOMAIN**: en caso de un dominio NIS.
- **NETWORKING\_IPV6** : activación o no del soporte de IPv6.

## b. Máquinas de tipo Debian

El fichero de configuración de las interfaces de redes en Debian (y Ubuntu) se sitúa en /etc/network/interfaces. No tiene el mismo formato que en Red Hat.

```
# cat interfaces
auto lo eth0 eth1
iface lo inet loopback

iface eth0 inet static
    address 192.161.1.60
    netmask 255.255.255.0
    broadcast 192.168.1.255
    gateway 192.168.1.1

iface eth1 inet dhcp
```

Este ejemplo muestra tres tipos de interfaces:

- la interfaz lo de loopback,
- la interfaz eth1 en dhcp, que no necesita una configuración más avanzada,
- la interfaz eth0 configurada de manera estática.

La sintaxis general de una declaración es la siguiente:

interfaz nombre tipo modo

Con una configuración estática, especifique los diferentes parámetros con las palabras claves siguientes:

- **address**: la dirección IP.
- **netmask**: la máscara de subred.
- **broadcast**: la dirección de broadcast.
- **gateway**: la pasarela por defecto.

La línea **auto** indica las interfaces que se activarán automáticamente en el arranque.

Para IPv6 la configuración es idéntica: basta con reemplazar inet con inet6.

El fichero **/etc/hostname** contiene el nombre de la máquina:

```
# cat /etc/hostname  
slyserver
```

### c. Encaminamiento

Con la utilización de los ficheros y comandos anteriores, no hace falta crear un encaminamiento específico, puesto que la pasarela por defecto está ya presente (vea parámetros generales) e **ifup** implementa automáticamente las rutas para las interfaces. Sin embargo, se puede utilizar el comando **route**.

Visualiza las rutas actuales:

```
route  
netstat -nr
```

En el ejemplo siguiente, las interfaces vmnet1 y vmnet8 son interfaces virtuales procedentes de la configuración de red de VMWare.

```
# netstat -rn  
Tabla de encaminamiento IP del núcleo  
Destino      Pasarela      Genmask      Indic      MSS  Ventana irtt  Iface  
192.168.211.0  0.0.0.0  255.255.255.0  U          0 0        0  vmnet8  
192.168.1.0   0.0.0.0  255.255.255.0  U          0 0        0  eth0  
172.16.248.0  0.0.0.0  255.255.255.0  U          0 0        0  vmnet1  
169.254.0.0   0.0.0.0  255.255.0.0    U          0 0        0  eth0  
127.0.0.0     0.0.0.0  255.0.0.0     U          0 0        0  lo  
0.0.0.0       192.168.1.1 0.0.0.0     UG         0 0        0  eth0
```

Creación de la entrada loopback:

```
route add -net 127.0.0.0
```

Crea la ruta hacia la red 192.168.1.0, que pasa por eth0. Se puede omitir netmask.

```
route add -net 192.168.1.0 netmask 255.255.255.0 eth0
```

Crea la pasarela por defecto hacia el enrutador:

```
route add default gw 192.168.1.254
```

Suprime la ruta hacia la red 172.16.0.0:

```
route del -net 172.16.0.0 eth0
```

La gestión de rutas IPv6 es idéntica, pero indicando el tipo de dirección usado al comando con la opción -A:

```
route -A inet6 add <ipv6> gw <ip6>
```

## 4. Herramientas de red

### a. FTP

Resulta útil conocer el comando **ftp** (*file transfer protocol*). Permite la transferencia de ficheros entre dos

máquinas. Coge como parámetro el nombre de la máquina remota. Para que el comando **ftp** funcione, el servicio ftp debe funcionar en la máquina remota y en el puerto 21.

Veamos un ejemplo (poco práctico) de conexión con error y nuevo intento.

```
ftp> open  
(to) machine  
Connected to machine.  
220 machine FTP server (Digital UNIX Version 5.60) ready.  
Name (machine:root): root  
331 Password required for root.  
Password:  
530 Login incorrect.  
Login failed.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> user  
(username) root  
331 Password required for root.  
Password:  
230 User root logged in.  
ftp> pwd  
257 "/" is current directory.
```

Sin embargo, lo más sencillo es:

```
$ ftp machine  
Connected to machine.  
220 machine FTP server (Digital UNIX Version 5.60) ready.  
Name (machine:root): root  
331 Password required for root.  
Password:  
230 User root logged in.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>
```

Veamos una lista de comandos **ftp**.

Comando	Acción
open	Seguido de un nombre de máquina, abre una conexión en la máquina especificada.
user	Inserción del usuario remoto para una conexión.
quit	Fin de la conexión y del comando ftp.
ascii	Transferencia de los ficheros en modo ASCII (conversión de los caracteres especiales y fin de línea en MS y Unix, por ejemplo).
binary	Transferencia de los ficheros en modo binario.
glob	Suprime la interpretación de los caracteres especiales.
help	Muestra la ayuda.

prompt	Seguido de on u off, activa o desactiva la confirmación individual de transferencia para cada fichero (mget o mput).
pwd	Muestra el directorio remoto actual.
cd	Seguido de la ruta, desplazamiento en el árbol remoto.
ls	Lista los ficheros de la máquina remota.
delete	Seguido de un nombre de fichero, suprime el fichero remoto.
mdelete	Múltiple. Suprime los ficheros remotos.
get	Recupera el fichero remoto.
mget	Múltiple. Recupera los ficheros remotos (lista o modelo).
put	Manda el fichero local hacia la máquina remota.
mput	Múltiple. Manda los ficheros locales a la máquina remota (lista o modelo).
close/disconnect	Cierra la sesión actual.
lcd	Cambia de directorio en la máquina local.
hash	Durante la transferencia, escribe una «#» en pantalla para cada buffer transferido.
system	Información sobre el sistema remoto.
recv	Recepción de un fichero.
send	Envío de un fichero.
rename	Renombra un fichero remoto.
mkdir	Crea un directorio en la máquina remota.
rmdir	Suprime un directorio en la máquina remota.
!comando	Ejecuta el comando local.

## b. Telnet

**Telnet** es un cliente ligero que permite abrir una conexión y una sesión en una máquina remota que propone un servidor telnet. A menudo, se inicia este servidor desde xinetd o inetd. Su sintaxis es muy simple:

```
$ telnet -l user machine port
```

```
Ejemplo:  
# telnet 192.168.1.60  
Trying 192.168.1.60...  
Connected to 192.168.1.60.  
Escape character is '^]'.  
Welcome to openSUSE 10.3 (i586) - Kernel 2.6.24.4-default (3).  
  
slyserver login: seb  
Contraseña:  
Última conexión: jueves 15 de mayo de 2008 a 06:26:30 CEST de consola  
en:0  
Tiene un nuevo mensaje.  
Have a lot of fun...  
seb@slyserver:~>
```

 ¡Cuidado! El servicio (y el cliente) telnet no tiene ningún tipo de seguridad: las conexiones transitan sin cifrar por la red, y cualquier esnifador IP (wireshark, por ejemplo) puede interceptar y ver todo lo que se hace. Incluso se transmite la contraseña sin cifrar. Evite utilizar este servicio y céntrese en OpenSSH.

### c. Ping

El comando **ping** es un comando central, incluso ineludible. Para saber si una máquina está accesible o no, lo primero que se suele hacer es intentar «hacer ping» (con la condición de que la configuración del firewall autorice las peticiones ICMP). El comando **ping6** hace lo mismo para una dirección IPv6.

Ping emite un «eco» de red, un poco como un sónar, y espera una respuesta, el retorno del eco. Para ello, utiliza el protocolo ICMP. Interrumpa el comando **ping** con [Ctrl] **C**.

```
$ ping www.kde.org  
PING www.kde.org (62.70.27.118) 56(84) bytes of data.  
64 bytes from 62.70.27.118: icmp_seq=1 ttl=57 time=10.5 ms  
64 bytes from 62.70.27.118: icmp_seq=2 ttl=57 time=11.3 ms  
64 bytes from 62.70.27.118: icmp_seq=3 ttl=57 time=10.4 ms  
64 bytes from 62.70.27.118: icmp_seq=4 ttl=57 time=11.5 ms  
...
```

Tres parámetros deben llamarle la atención:

- **-c** permite especificar el número de ecos que se deben emitir.
- **-b** permite emitir un eco en una dirección de broadcast.
- **-I** permite especificar la interfaz de red.

En el primer caso, el parámetro puede ser útil en un script para probar que un servidor responde:

```
# ping -c 1 10.9.238.170 >/dev/null 2>&1 && echo "El servidor contesta"  
El servidor contesta
```

En el segundo caso, todas las direcciones de la subred correspondientes a la dirección de broadcast deben responder.

```
# ping -b 192.168.1.255  
WARNING: pinging broadcast address  
PING 192.168.1.255 (192.168.1.255) 56(84) bytes of data.  
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.232 ms  
64 bytes from 192.168.1.60: icmp_seq=1 ttl=64 time=0.240 ms
```

```
64 bytes from 192.168.1.130: icmp_seq=1 ttl=255 time=0.285 ms
64 bytes from 192.168.1.139: icmp_seq=1 ttl=255 time=0.292 ms
...
```

En el último caso, puede especificar una tarjeta de salida. Esta opción es muy útil para comprobar una resolución DNS o una ruta.

```
# ping -I eth0 192.168.1.60
PING 192.168.1.60 (192.168.1.60) from 192.168.1.10:eth0: 56(84) bytes
of data.
64 bytes from 192.168.1.60: icmp_seq=1 ttl=62 time=0.478 ms
64 bytes from 192.168.1.60: icmp_seq=2 ttl=62 time=0.408 ms
...
```

## d. Traceroute

Cuando intenta acceder a un anfitrión remoto desde su máquina, los paquetes IP pasan a menudo por muchas rutas, a veces diferentes según el punto de partida y de destino, los cuellos de botella, etc. El trayecto pasa por numerosas pasarelas (gateways), que dependen de las rutas por defecto o predefinidas de cada una de ellas.

El comando **traceroute** permite visualizar cada uno de los puntos de paso de sus paquetes IP a su destino en un anfitrión dado. En el ejemplo siguiente, el anfitrión ubicado en la región parisina en la red del proveedor Free intenta determinar la ruta recorrida para ir al servidor [www.kde.org](http://www.kde.org). Se enmascarará la dirección IP fuente (fuera de la red local). El comando **traceroute6** es idéntico para IPv6.

```
$ traceroute www.kde.org
traceroute to www.kde.org (62.70.27.118), 30 hops max, 40 byte packets
 1 DD-WRT (192.168.1.1)  0.558 ms   0.533 ms   0.585 ms
 2 82.xxx.yyy.zzz (82.xxx.yyy.zzz)  6.339 ms   6.404 ms   6.901 ms
 3 * * *
 4 * * *
 5 212.73.205.5 (212.73.205.5)  39.267 ms   35.499 ms   31.736 ms
 6 ae-12-55.car2.Paris1.Level3.net (4.68.109.144)  6.485 ms
ae-22-52.car2.Paris1.Level3.net (4.68.109.48)  6.401 ms   6.338 ms
 7 UUnet-Level3.Level3.net (212.73.240.206)  6.113 ms   6.152 ms
5.866 ms
 8 so-3-2-0.TL2.PAR2.ALTER.NET (146.188.8.121)  6.107 ms   6.410 ms
6.365 ms
 9 so-2-2-0.TL2.STK2.ALTER.NET (146.188.7.33)  87.323 ms   86.840 ms
87.010 ms
10 so-7-1-0.XR2.OSL2.ALTER.NET (146.188.15.62)  96.491 ms   97.148 ms
96.488 ms
11 ge-0-1-0.GW6.OSL2.ALTER.NET (146.188.3.242)  95.972 ms   95.934 ms
96.108 ms
12 213.203.63.74 (213.203.63.74)  95.320 ms   94.321 ms   96.188 ms
13 leeloo.troll.no (62.70.27.10)  94.064 ms   94.052 ms   92.374 ms
14 jamaica.kde.org (62.70.27.118)  97.064 ms   96.182 ms   97.853 ms
```

## e. Whois

¿Sabía que puede obtener toda la información que deseé sobre un dominio ([pepito.es](http://pepito.es)) usando el comando **whois**? Por ejemplo, para obtener toda la información sobre el dominio [kde.org](http://kde.org):

```
> whois kde.org
...
Domain ID:D1479623-LR0R
Domain Name:KDE.ORG
Created On:14-Dec-1996 05:00:00 UTC
Last Updated On:12-Oct-2007 13:10:18 UTC
```

```

Expiration Date:13-Dec-2012 05:00:00 UTC
Sponsoring Registrar:easyDNS Technologies Inc. (R1247-LR0R)
Status:CLIENT TRANSFER PROHIBITED
Status:CLIENT UPDATE PROHIBITED
Registrant ID:tu2YDGaiunEvz5QA
Registrant Name:Trolltech AS
Registrant Organization:Trolltech AS
Registrant Street1:Sandakerveien 116, PO Box 4332 Nydalen
Registrant Street2:
Registrant Street3:
Registrant City:Oslo
Registrant State/Province:N/A
Registrant Postal Code:N-0402
Registrant Country:NO
Registrant Phone:+1.4721604800
Registrant Phone Ext.:
Registrant FAX:+1.4721604801
Registrant FAX Ext.:
Registrant Email:hostmaster@trolltech.com
Admin ID:tubEUVkFfutkJZMD
Admin Name:Trolltech AS
Admin Organization:Trolltech AS
Admin Street1:Sandakerveien 116, PO Box 4332 Nydalen
Admin Street2:
Admin Street3:
Admin City:Oslo
Admin State/Province:N/A
Admin Postal Code:N-0402
Admin Country:NO
Admin Phone:+1.4721604800
Admin Phone Ext.:
Admin FAX:+1.4721604801
Admin FAX Ext.:
Admin Email:hostmaster@trolltech.com
Tech ID:tuSfXVVJtggMdKwM
Tech Name:Trolltech AS
Tech Organization:Trolltech AS
Tech Street1:Sandakerveien 116, PO Box 4332 Nydalen
Tech Street2:
Tech Street3:
Tech City:Oslo
Tech State/Province:N/A
Tech Postal Code:N-0402
Tech Country:NO
...

```

## f. Netstat

El comando **netstat** permite obtener una multitud de información sobre la red y los protocolos.

El parámetro **-i** permite obtener el estado de las tarjetas de redes con el fin de determinar una posible avería o un problema de cable:

```

# netstat -i
Tabla de interfaces del núcleo
Iface MTU Met RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0 1500 0 2332007 0 0 0 677842 0 0 0 BMRU
lo 16436 0 1109 0 0 0 1109 0 0 0 LRU

```

Si

añade el parámetro **-e**, obtiene el mismo resultado que con **ifconfig -a**.

```
# netstat -ei
```

El

```

Tabla de interfaces del núcleo
eth0  Vínculo encap:Ethernet HWaddr 00:XX:D3:XX:AA:XX
      inet adr:12.168.1.60 Bcast:192.168.1.255 Máscara:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:2335314 errors:0 dropped:0 overruns:0 frame:0
          TX packets:678095 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:1055212145 (1006.3 MB) TX bytes:61264196 (58.4 MB)
          Interruption:20 Dirección basica:0x8c00

lo   Vínculo encap:Bucle local
      inet adr:127.0.0.1 Máscara:255.0.0.0
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:1109 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1109 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:60423 (59.0 Kb) TX bytes:60423 (59.0 KB)

```

parámetro **-r** permite obtener las tablas de encaminamiento como ruta. Añada el parámetro **-n** para indicar las IP en lugar de los nombres.

```

# netstat -rn
Tabla de encaminamiento IP del núcleo
Destino     Pasarela     Genmask     Indic     MSS Ventana irtt
Iface
192.168.211.0  0.0.0.0    255.255.255.0  U         0 0        0
vmnet8
192.168.1.0    0.0.0.0    255.255.255.0  U         0 0        0
eth0
172.16.248.0   0.0.0.0    255.255.255.0  U         0 0        0
vmnet1
169.254.0.0    0.0.0.0    255.255.0.0   U         0 0        0
eth0
127.0.0.0      0.0.0.0    255.0.0.0    U         0 0        0
lo
0.0.0.0        192.168.1.1  0.0.0.0     UG        0 0        0
eth0

```

parámetro **-a** permite visualizar todas las conexiones, para todos los protocolos, incluido los puertos en escucha de la máquina. La salida es demasiado larga para reproducirla en estas páginas.

```
# netstat -a | wc -l
495
```

parámetro **-A** permite especificar el protocolo que es preciso consultar: **inet**, **inet6** (**ipv6**), **unix**, **ipx**, **ax25**, **netrom** y **ddp**.

```

# netstat -a -A inet
Conexiones Internet activas (servidores y establecidas)
Proto Recv-Q Send-Q Dirección local           Dirección remota
Estado
tcp      0      0 localhost:716              *:*
LISTEN
tcp      0      0 *:sunrpc                *:*
LISTEN
tcp      0      0 localhost:ipp             *:*
LISTEN
tcp      0      0 localhost:smtp             *:*
LISTEN
tcp      0      0 localhost:hpssd            *:*
LISTEN

```

tcp	0	0	slyserver:41851	imap.tele2.es:imap
ESTABLISHED				
tcp	0	0	slyserver:41850	imap.tele2.es:imap
ESTABLISHED				
tcp	0	0	slyserver:54220	by1msg4176111.gate:msnp
ESTABLISHED				
tcp	0	0	slyserver:34267	by2msg2105007.phx.:msnp
ESTABLISHED				
tcp	0	0	slyserver:47990	by1msg4082314.phx.:msnp
ESTABLISHED				
udp	0	0	*:filenet-tms	*:*
udp	0	0	*:mdns	*:*
udp	0	0	*:sunrpc	*:*
udp	0	0	*:ipp	*:*
udp	0	0	172.16.248.1:ntp	*:*
udp	0	0	192.168.211.1:ntp	*:*
udp	0	0	slyserver:ntp	*:*
udp	0	0	localhost:ntp	*:*
udp	0	0	*:ntp	*:*
raw	0	0	*:icmp	*:*

7

para terminar, el parámetro **-p** permite indicar, cuando sea posible, el ID y el nombre del proceso.

```
# netstat -A inet -p
Conexiones a Internet activas (sin servidores)
Proto Recv-Q Send-Q Dirección local           Dirección remota
Estado      PID/Program name

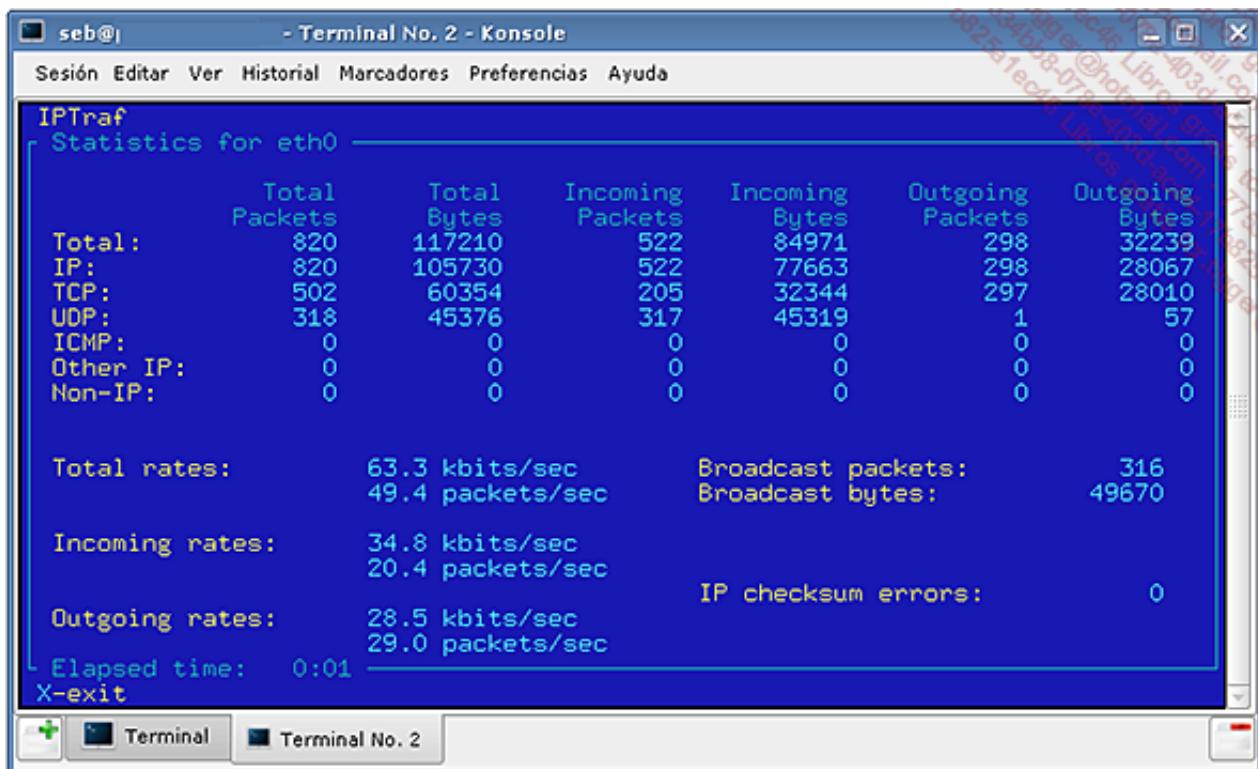
...
tcp        0      0  slyserver:54220          by1msg4176111.gate:msnp
ESTABLISHED 4041/kopete
tcp        0      0  slyserver:34267          by2msg2105007.phx.:msnp
ESTABLISHED 4041/kopete
tcp        0      0  slyserver:47990          by1msg4082314.phx.:msnp
ESTABLISHED 4041/kopete
```

## g. IPTraf

El comando **iptraf** permite visualizar en tiempo real la actividad de la red mediante una herramienta de texto, opcionalmente interactiva (línea de comandos). Los menús son claros. Puede moverse por ellos con las teclas de dirección y los diferentes atajos especificados.

La captura siguiente muestra la visualización detallada de las estadísticas de la tarjeta **eth0**. Esta pantalla está accesible mediante la línea de comandos con:

```
# iptraf -d eth0
```



*IPTraf analiza el tráfico por eth0.*

## 5. Ficheros generales

### a. /etc/resolv.conf

Se utiliza el fichero `/etc/resolv.conf` para indicar al sistema qué servidores de nombres y qué dominios hay que consultar para resolver las peticiones DNS clientes. Estas API, al igual que las API estándares de Linux, están incluidas en la librería (no hace falta añadir herramientas adicionales). Esta librería se llama `resolver`.

---

► Al configurar DHCP en principio se actualiza automáticamente este fichero y no se debería modificar, salvo que se prohíba la configuración DNS en el cliente.

---

```
$ cat /etc/resolv.conf
domain midominio.org
search midominio.org
nameserver 192.168.1.1
nameserver 192.168.1.2
```

**domain:** nombre del dominio local. Las peticiones se suelen reducir a unos atajos relativos hacia el dominio local. Si no está creado, se debe determinar el nombre del dominio a partir del nombre completo del anfitrión: corresponde a la parte ubicada después del primer «.».

- **search:** lista de los dominios de búsqueda. Por defecto, durante la utilización de atajos (nombres de anfitriones cortos) el resolver inicia una búsqueda sobre el dominio definido por la línea `domain`, pero se puede especificar aquí una lista de dominios separados por espacios o comas.
- **nameserver:** dirección IP del servidor de nombres (el servidor DNS). Se puede colocar un máximo

de tres. El resolver intenta utilizar el primero. En caso de fracaso (timeout), pasa al segundo, y así sucesivamente.

- **opciones:** se pueden especificar opciones. Por ejemplo, **timeout:n**, donde n (en segundos) indica el tiempo de espera de respuesta de un servidor de nombres antes de pasar al siguiente.

## b. /etc/hosts y /etc/networks

Sin siquiera utilizar un servidor de nombres, es posible establecer una correspondencia entre las direcciones IP y los nombres de las máquinas dentro del fichero **/etc/hosts**.

```
192.168.1.1    server1 www1 ftp
192.168.1.11   puesto1
192.168.1.12   puesto2
```

Puede hacer lo mismo para nombrar las redes (lo que puede ser útil para los **tcp\_wrappers** o el comando **route**) en el fichero **/etc/networks**.

```
loopnet    127.0.0.0
localnet   192.168.1.0
```

## c. /etc/nsswitch.conf

El fichero **/etc/nsswitch.conf** permite determinar el orden en el cual el resolver (u otros servicios) recupera su información. Las dos líneas en negrita en el ejemplo indican que, durante una petición de resolución de nombre (o red), los ficheros son prioritarios. Primero se lee el fichero **/etc/hosts**; luego, si el resolver no encuentra la información, busca mediante una resolución DNS.

```
passwd: compat
group: compat

hosts:      files dns
networks:   files dns

services:   files
protocols:  files
rpc:        files
ethers:     files
netmasks:   files
netgroup:   files nis
publickey:  files

bootparams: files
automount:  files nis
aliases:    files
```

 Puede ocurrir que determinados programas no utilicen el resolver sino directamente el DNS o el fichero **etc/hosts**, o inviertan el orden establecido en **/etc/nsswitch.conf**. En este caso, no es posible prever (ni predecir) el funcionamiento correcto de este tipo de programas...

## d. /etc/services

El fichero **/etc/services** contiene la lista de los servicios de redes conocidos de Unix, así como los puertos y protocolos asociados. Muchos servicios (entre los cuales se halla **xinetd**) y subsistemas, como el firewall de Linux, lo utilizan.

# Servicios redes xinetd

## 1. Presentación

El demonio **xinetd** es un «superservicio» que permite controlar el acceso a un conjunto de servicios, **telnet** por ejemplo. Se pueden configurar muchos servicios de redes para funcionar con **xinetd**, como los servicios ftp, ssh, samba, rcp, http, etc. Se pueden aplicar opciones de configuración específicas para cada servicio gestionado.

Cuando un cliente se conecta a un servicio de red controlado por **xinetd**, xinetd recibe la petición y verifica primero las autorizaciones de acceso TCP (vea **tcp\_wrappers** en el próximo capítulo); luego, las reglas definidas para este servicio (autorizaciones específicas, recursos asignados, etc.). El demonio levanta una instancia del servicio y le cede la conexión. A partir de entonces, **xinetd** ya no interfiere en la conexión entre el cliente y el servidor.

## 2. Configuración

Los ficheros de configuración son:

- **/etc/xinetd.conf**: configuración global
- **/etc/xinetd.d/\***: directorio que contiene los ficheros específicos para los servicios. Existe un fichero por servicio, con el mismo nombre que el especificado en **/etc/services**.

```
$ ls -l /etc/xinetd.d
total 92
-rw-r--r-- 1 root root 313 sep 22 2007 chargen
-rw-r--r-- 1 root root 333 sep 22 2007 chargen-udp
-rw-r--r-- 1 root root 256 mar 20 22:11 cups-lpd
-rw-r--r-- 1 root root 409 nov  4 2005 cvs
-rw-r--r-- 1 root root 313 sep 22 2007 daytime
-rw-r--r-- 1 root root 333 sep 22 2007 daytime-udp
-rw-r--r-- 1 root root 313 sep 22 2007 discard
-rw-r--r-- 1 root root 332 sep 22 2007 discard-udp
-rw-r--r-- 1 root root 305 sep 22 2007 echo
-rw-r--r-- 1 root root 324 sep 22 2007 echo-udp
-rw-r--r-- 1 root root 492 sep 22 2007 netstat
-rw-r--r-- 1 root root 207 abr 23 19:04 rsync
-rw-r--r-- 1 root root 337 feb 17 14:22 sane-port
-rw-r--r-- 1 root root 332 sep 22 2007 servers
-rw-r--r-- 1 root root 334 sep 22 2007 services
-rw-r--r-- 1 root root 351 jun 21 2007 svnservice
-rw-r--r-- 1 root root 277 nov  8 2007 swat
-rw-r--r-- 1 root root 536 sep 21 2007 systat
-rw-r--r-- 1 root root 387 feb  4 10:11 tftp.rpmsave
-rw-r--r-- 1 root root 339 sep 22 2007 time
-rw-r--r-- 1 root root 333 sep 22 2007 time-udp
-rw-r--r-- 1 root root 2304 abr  4 11:39 vnc
-rw----- 1 root root 768 sep 22 2007 vsftpd
```

Contenido de xinetd.conf:

```
defaults
{
    instances          = 60
    log_type          = SYSLOG authpriv
```

```

    log_on_success      = HOST PID
    log_on_failure     = HOST
    cps                = 25 30
}
includedir /etc/xinetd.d

```

**instances:** número máximo de peticiones que un servicio **xinetd** puede gestionar en un instante dado.

- **log\_type:** en nuestro caso, el demonio **syslog** gestiona las trazas mediante **authpriv** y las trazas están colocadas en `/var/log/secure`. **FILE /var/log/xinetd** hubiera colocado las trazas en `/var/log/xinetd`.
- **log\_on\_success:** **xinetd** va a registrar el evento si la conexión al servicio tiene éxito. La información trazada son el cliente (**HOST**) y el **ID** del proceso servidor que trata de la conexión.
- **log\_on\_failure:** ídem para los fracasos. Resulta fácil saber qué clientes han intentado conectarse si, por ejemplo, no se autoriza la conexión.
- **cps:** **xinetd** sólo autoriza 25 conexiones por segundo a un servicio. Si se alcanza el límite, **xinetd** esperará 30 segundos antes de autorizar de nuevo las conexiones.
- **includedir:** incluye las opciones de los ficheros presentes en el directorio indicado.

#### Ejemplo /etc/xinetd.d/telnet:

```

# default: on
# description: The telnet server serves telnet sessions; it uses \
#               unencrypted username/password pairs for authentication.
service telnet
{
    disable          = no
    flags            = REUSE
    socket_type     = stream
    wait             = no
    user             = root
    server           = /usr/sbin/in.telnetd
    log_on_failure   += USERID
}

```

La primera línea de comentario, **default**, tiene una importancia particular. No la interpreta **xinetd**, sino **ntsysv** o **chkconfig**, para determinar si el servicio está activo.

- **service:** nombre del servicio que corresponde a un servicio definido en `/etc/services`.
- **flags:** atributos para la conexión. **REUSE** indica que se volverá a utilizar el socket para una conexión telnet.
- **socket\_type:** especifica el tipo de socket. En general, **stream** (tcp) o **dgram** (udp). Una conexión directa IP se hace por **raw**.
- **wait:** indica si el servidor es single-threaded (yes) o multi-threaded (no).
- **user:** con qué cuenta de usuario se iniciará el servicio.
- **server:** ruta del ejecutable que se debe iniciar.
- **log\_on\_failure:** el **+=** indica que se añade la opción asociada al fichero de traza, además de las opciones por defecto. Aquí: el login.
- **disable:** indica si el servicio está activo o no.

Algunas opciones pueden mejorar las condiciones de acceso y la seguridad:

- **only\_from**: permite el acceso únicamente a los anfitriones especificados.
- **no\_access**: impide el acceso a los anfitriones especificados (p. ej.: 172.16.17.0/24).
- **access\_times**: autoriza el acceso únicamente en una franja horaria dada (p.ej.: 09:00-18:30).

### 3. Inicio y parada de los servicios

Se distinguen dos casos.

**Primer caso**, el servicio **xinetd** es un servicio como otro cualquiera cuyo inicio o cuya parada puede efectuarse con el comando **service** o directamente mediante la ejecución de /etc/init.d/xinetd.

```
# service xinetd start
```

En este caso, el comando **chkconfig** (Red Hat, openSUSE) autoriza o no el arranque del servicio al inicio para cada nivel de ejecución (runlevel).

```
# chkconfig --level 345 xinetd on
```

**Segundo caso**, como **xinetd** gestiona varios servicios, la parada de xinetd detiene todos los servicios asociados y el lanzamiento de **xinetd** inicia todos los servicios asociados. No es posible elegir qué servicios de **xinetd** se han iniciado en tal o cual nivel de ejecución. Pero puede elegir entre activar o desactivar simplemente un servicio con **chkconfig**.

```
# chkconfig telnet on
```

# Conexión PPP

## 1. Elección y configuración del módem

### a. El caso de los Winmodems

Linux soporta completamente todos los módems RTC (analógicos) que se conectan en un puerto serie (externo) o que emulan un verdadero puerto serie (tarjeta PCI o mediante el puerto USB).

Sin embargo, existe una categoría particular de módems llamados los **winmodems**. Se presentan a veces como «verdaderos» módems (en ocasiones se parecen). En general, huya de este tipo de módems. No obstante, algunos modelos son soportados por Linux. Para obtener información sobre este tema, diríjase al sitio <http://linmodems.org/>

Linux reconoce como módems otros adaptadores distintos de los módems RTC. Es el caso de algunos adaptadores ADSL, pero también de los teléfonos móviles conectados por un cable USB o mediante una conexión Bluetooth.

### b. Los ficheros periféricos

Los puertos serie de tipo RS232 se llaman **ttySn**:

- **/dev/ttys0**: primer puerto serie.
- **/dev/ttys1**: segundo puerto serie.
- etc.

Los puertos serie de tipo USB se llaman **ttyUSBn**: **/dev/ttys0**, y así sucesivamente.

Los puertos serie de comunicación mediante bluetooth se llaman **rfcommn** (para radio frequency communication): **/dev/rfcomm0**, y así sucesivamente.

Para utilizar los puertos serie y establecer una comunicación, debe poder escribir en los periféricos (para mandar órdenes) y, así, o bien conocer los derechos correspondientes, o bien utilizar un programa SUID, o aun disponer de reglas udev adaptadas.

### c. Ajustar el puerto serie

Los puertos serie se gestionan mediante el comando **setserial**.

El comando **setserial** permite consultar la configuración de un puerto serie con el parámetro **-g**. El puerto serie ttys0 es de tipo 16550A (el más rápido), utiliza el IRQ 4 y el puerto de dirección 0x03f8.

```
# setserial -g /dev/ttys0  
/dev/ttys0, UART: 16550A, Port: 0x03f8, IRQ: 4
```

Como la información no es suficiente, añada el parámetro **-a**. Obtiene entre otras cosas la velocidad de la línea, que aquí es de 115200 bits por segundo (se debe proscribir la palabra «baudio»).

```
# setserial -a -g /dev/ttys0  
/dev/ttys0, Line 0, UART: 16550A, Port: 0x03f8, IRQ: 4  
      Baud_base: 115200, close_delay: 50, divisor: 0  
      closing_wait: 3000  
      Flags: spd_normal skip_test
```

Setserial permite también configurar el puerto. Al consultar el puerto, puede saber qué parámetros han permitido su ajuste con -G:

```
# setserial -G /dev/ttyS0
/dev/ttyS0 uart 16550A port 0x03f8 irq 4 baud_base 115200 spd_normal
skip_test
```

De ahí es posible extraer nuevos valores. Por ejemplo, pase el puerto serie a una velocidad de 57600 bps:

```
# setserial /dev/ttyS0 baud_base 57600
# setserial -a -g /dev/ttyS0
/dev/ttyS0, Line 0, UART: 16550A, Port: 0x03f8, IRQ: 4
    Baud_base: 57600, close_delay: 50, divisor: 0
    closing_wait: 3000
    Flags: spd_normal skip_test
```

#### d. Los comandos AT

Todos los módems utilizan un par de comandos estándares llamados **comandos AT**. Su verdadero nombre es **Comandos Hayes**, del nombre de la empresa que los inventó. AT significa Attention (Cuidado). Después de estas primeras letras, el módem espera una serie que permita configurarlo, numerarlo, colgarlo, etc.

El par suele ser estándar, pero la configuración cambia de un modelo a otro. Si no quiere entrar en detalles, una configuración genérica es suficiente y funciona para la casi totalidad de los módems. Como no es posible describir los comandos AT, encontrará una lista en el sitio 3com (que ha comprado US Robotics, el mejor fabricante de módems): <http://www.usr.com/support/3cp3056/3cp3056-spanish-ug/atcoms.htm>.

Sin embargo, veamos algunos:

- Marcar: ATDT0102030405
- Contestar: ATA
- Colgar: ATH

## 2. PPP

El protocolo **PPP** (*Point to Point Protocol*) le permite relacionarse con otra máquina con el fin de acceder a ella y a su red, lo que suele ser el caso de Internet. Hoy en día, y a pesar de las numerosas soluciones propuestas por el cable o el ADSL, algunas conexiones se siguen haciendo por módem RTC (módem clásico conectado en un puerto USB, serie o interno) conectado a un enchufe de los de siempre.

Para establecer una conexión PPP necesita:

- Un cliente que disponga de las herramientas ppp (pppd) y chat para dialogar con el servidor.
- Un servidor que disponga de pppd y medios para facilitar una dirección IP (dhcp).
- Un módem.

Lo que viene a continuación no considera la parte cliente.

Debe conocer:

- El puerto serie en el cual está conectado su módem: ttySX (serie o USB), ttyACMX (USB), rfcommX (Bluetooth), etc.
- El número de teléfono de su proveedor de acceso a Internet (PAI).

- El número de usuario y la contraseña en su PAI.
- La dirección del servidor DNS de su PAI.

 El módem no tiene por qué ser RTC. Un móvil reconocido como módem mediante el cable de conexión al PC o con el protocolo Bluetooth sirve perfectamente de módem. Para que sea conforme a las normas 3G o Edge, las flujo pueden ser muy impresionantes. Un gran número de conexiones ADSL se efectúan también mediante el protocolo PPP. En este caso, la serie se aplica, pero se deben prever unas modificaciones.

### 3. Conexión mediante la consola

#### a. Manualmente

En el ejemplo siguiente:

- El número de teléfono es 0102030405.
- El login es «login».
- La contraseña es «password».
- El periférico es /dev/modem.

Aunque DHCP pueda gestionarlo en el momento de la conexión, puede ser necesario modificar el fichero /etc/resolv.conf para indicar los servidores de nombres (DNS) de su proveedor.

La conexión PPP necesita que el servicio **pppd** (en general, /usr/sbin/pppd) sea ejecutado como root. Para ello, se suele ubicar el derecho SUID:

```
-rwsr-xr-t 1 root root 316392 2008-04-04 19:03 pppd*
```

Otra solución es dar estos derechos a la herramienta de conexión (chat, kppp, etc.) o modificar en consecuencia los derechos de los periféricos (es el caso de varias distribuciones).

Se sitúan los ficheros de configuración en /etc/ppp:

```
# ls -l /etc/ppp
total 48
-rw----- 1 root root 690 sep 21 2007 chap-secrets
-rw-r--r-- 1 root root 449 sep 21 2007 filters
lrwxrwxrwx 1 root root      5 may   9 20:47 ip-down -> ip-up
drwxr-xr-x 2 root root 4096 sep 21 2007 ip-down.d
-rw xr-xr-x 1 root root 6175 abr 24 00:26 ip-up
drwxr-xr-x 2 root root 4096 sep 21 2007 ip-up.d
-rw-r--r-- 1 root root 7943 sep 21 2007 options
-rw----- 1 root root 340 sep 21 2007 options.pptp
-rw----- 1 root root 1219 sep 21 2007 pap-secrets
drwxr-xr-x 2 root root 4096 feb 23 23:18 peers
-rw xr-xr-x 1 root root 3778 abr 24 00:26 poll.tcpip
```

Veamos un ejemplo de conexión a un servidor PPP:

```
#!/bin/sh
/usr/sbin/pppd connect '/usr/sbin/chat -v ABORT ERROR ABORT "NO CARRIER" \
ABORT BUSY "" ATZ OK ATDT0102030405 CONNECT "" ogin: "login" \
word: "password" \'
```

```
/dev/modem 38400 noipdefault debug crtscts modem defaultroute &
```

## b. Por los ficheros

Va a necesitar dos ficheros. El primero contendrá los comandos del servicio pppd; el segundo, la secuencia de comunicación con el PAI. Los dos están en /etc/ppp/peers.

Veamos el primer fichero /etc/ppp/peers/cnx1:

```
# cat /etc/ppp/peers/cnx1
/dev/modem
connect '/usr/sbin/chat -v -f /etc/ppp/peers/cnx1-chat'
defaultroute
noipdefault
usepeerdns
115200

debug
noauth

maxfail 10
lcp-echo-interval 5
lcp-echo-failure 12
holdoff 3
noaccomp noccp nobsdcomp nodeflate nopcomp novj novjccomp
lock
crtscts
```

Cada línea contiene al menos una instrucción; presentamos las más pertinentes:

- **/dev/modem**: el periférico de conexión (el módem);
- **connect**: la cadena de conexión mandada al PAI;
- **defaultroute**: la ruta facilitada por el PAI sustituye a la ruta por defecto;
- **noipdefault**: el PAI provee el IP por su DHCP;
- **usepeerdns**: recupera la información DNS del PAI;
- **115200**: la velocidad de comunicación del periférico (se negociará);
- **debug**: provee el detalle completo de la conexión;
- **noauth**: no es el script ppp que establece la autenticación (ver línea connect);
- **maxfail**: n intentos de conexión antes de abandonar;
- **holdoff**: espera de n segundos entre dos conexiones;
- **lock**: permite el acceso exclusivo al fichero periférico;
- **crtscs**: activa el control de flujo material.

Veamos el segundo fichero, /etc/ppp/peers/cnx1-chat, utilizado por la línea **connex** del primer fichero,/etc/ppp/peers/cnx1:

```
# cat /etc/ppp/peers/cnx1-chat
ABORT ERROR
ABORT "NO CARRIER"
ABORT BUSY "" ATZ
OK ATDT0102030405
```

```
CONNECT ""  
ogin: "login"  
word: "password"
```

Sólo se trata de un ejemplo. Tiene que comprobar, tanto por parte de su PAI como en la documentación de su módem, cuáles son los comandos correctos AT que hay que pasar (suelen ser estándar).

### c. Conexión

Incialice la conexión:

```
# pppd call cnx1
```

Debería ver las leds de su módem parpadear y, si el altavoz está activado, se oye el pitido característico. Si se establece la conexión, una nueva interfaz de red aparece: **ppp0**.

```
# ifconfig ppp0  
ppp0      Link encap:Point-Point Protocol  
          inet addr:10.xx.yy.zz  P-t-P:10.xx.yy.zz Mask:255.255.255.0  
            UP POINTOPOINT RUNNING MTU:552 Metric:1  
            RX packets:0 errors:0 dropped:0 overruns:0  
            TX packets:0 errors:0 dropped:0 overruns:0
```

 Las distribuciones generalmente vienen acompañadas de herramientas de tipo Network Manager que permiten establecer una conexión ethernet, Wi-Fi o ppp en un instante con algunos clics. La distribución Mandriva merece una mención especial: la configuración de una conexión Internet mediante un móvil 3G a través del protocolo Bluetooth se hace en unos segundos, con la herramienta Drakconf que lo detecta todo por sí misma.

# OpenSSH

## 1. Presentación

OpenSSH es un protocolo de shell con prestaciones de seguridad, un mecanismo que permite la autenticación segura, la ejecución remota y la conexión a distancia. Permite también el transporte seguro del protocolo X Window. En realidad, es capaz encapsular protocolos no seguros redireccionando los puertos.

Los paquetes que se deben utilizar para un servidor son **openssh**, **openssl** y **openssh-clients**. Para X se añaden los paquetes **openssh-askpass\*** (puede haber varios en función del entorno de escritorio). La lista de los paquetes que hay que instalar depende de cada distribución.

El uso más común sigue siendo el acceso remoto seguro a una máquina mediante el cliente ssh.

## 2. Configuración

La configuración es `/etc/ssh/sshd_config`. Si es preciso, se pueden modificar algunas opciones:

- **Port**: el número de puerto por defecto es 22;
- **Protocol**: fijado en 2,1, autoriza SSH1 y SSH2. Se preferirá SSH2 y así se dejará el valor 2;
- **ListenAddress**: por defecto ssh escucha en todos los IP del servidor. Se puede autorizar únicamente la escucha en una interfaz dada;
- **PermitRootLogin**: ssh autoriza las conexiones de root. Se puede poner el valor a «**no**». En este caso, habrá que conectarse como simple usuario y pasar por **su** o **sudo**;
- **Banner**: ruta de un fichero cuyo contenido se mostrará a los usuarios durante la conexión.

ssh es un servicio System V que puede iniciarse con service o directamente por `/etc/init.d/sshd`.

```
# service sshd start
```

## 3. Utilización

El comando **ssh** permite establecer una conexión.

```
$ ssh -l login host
$ ssh login@host
```

La opción **-X** permite activar la redirección (forwarding) del protocolo X Window.

```
$ ssh -X login@host
```

## 4. Claves y conexión automática

Es posible establecer una conexión automática hacia otra máquina sin introducir una contraseña. Para ello, es necesario generar un par de claves, privada y pública, desde la cuenta de usuario del cliente (la máquina que se va a conectar). No se debe introducir ninguna frase de contraseña.

Por parte del servidor ssh, se debe colocar la clave pública del cliente en un fichero que contiene las claves

autorizadas a conectarse en la cuenta de destino.

### a. Lado cliente

- Genere una clave en formato RSA con el comando **ssh-keygen**:

```
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/bean/.ssh/id_rsa):
Created directory '/home/bean/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/bean/.ssh/id_rsa.
Your public key has been saved in /home/bean/.ssh/id_rsa.pub.
The key fingerprint is:
f6:39:23:4e:fa:53:d0:4e:65:7f:3f:fd:a3:f4:8e:2a bean@p64p17bicb3
```

- El directorio del usuario contiene ahora un directorio .ssh:

```
$ cd .ssh
$ ls
id_rsa  id_rsa.pub
```

- El fichero id\_rsa.pub contiene la clave pública:

```
$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAkB/VskR9v708J2EDG1LM1Q6HmKJcP2UenurnSr7rWTSZK5w9Hzn4DCz5iMzLAPc4659I0uKJbmF3vBXozIgLrCdCZCQEhhPLwJVLXbGNc8lMf742E/WqKKJ/uQYb31iPAU7Efosei+DVZ21No725Xj iSCZ2qzKKx7ZunQEtxW0eVkwvlA0u7Hvrwn+FQksW3NXwTxwHhudSw7S6kIC3tyF5rkzfku7zQb0GDGGPiF3a0vd0oSBNgijTz+M0PaoXXI3brMd66WkGfSwf4ofYKNDCA/3TQ4xU6WxkxqTBcsjEm1gIymFAyxDo+zzf63jxLG08Pp50DKf7DUqBx7+rjw==
bean@slyserver
```

### b. Lado servidor

- Vaya al directorio .ssh de la cuenta a la cual desea acceder en el servidor (cree una si no existe):  

```
$ cd /home/seb/.ssh
```
- Edite el fichero authorized\_keys2 (creélo si no existe) y copie dentro en una nueva línea el contenido del fichero id\_rsa.pub del cliente. Guarde.

```
$ echo "ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAkB/VskR9v708J2EDG1LM1Q6HmKJcP2UenurnSr7rWTSZK5w9Hzn4DCz5iMzLAPc4659I0uKJbmF3vBXozIgLrCdCZCQEhhPLwJVLXbGNc8lMf742E/WqKKJ/uQYb31iPAU7Efosei+DVZ21No725Xj iSCZ2qzKKx7ZunQEtxW0eVkwvlA0u7Hvrwn+FQksW3NXwTxwHhudSw7S6kIC3tyF5rkzfku7zQb0GDGGPiF3a0vd0oSBNgijTz+M0PaoXXI3brMd66WkGfSwf4ofYKNDCA/3TQ4xU6WxkxqTBcsjEm1gIymFAyxDo+zzf63jxLG08Pp50DKf7DUqBx7+rjw== bean@slyserver" >> authorized_keys2
```

- Intente una conexión; no se requiere la contraseña:

```
$ ssh seb@slyserver
```

# Montar un servidor DHCP

## 1. Presentación

El servicio **DHCP** (*Dynamic Host Configuration Protocol*), protocolo de configuración dinámico de los anfitriones, permite a los anfitriones de una red pedir y recibir información de configuración (dirección, encaminamiento, DNS, etc.). Suele haber un solo servidor DHCP por segmento de red incluso si varios son posibles. Si el servidor está en otro segmento, se puede utilizar un agente de retransmisión DHCP.

Dicho de otro modo, un cliente DHCP busca sólo un servidor DHCP, que le comunicará su dirección IP. Se asigna la dirección IP de manera dinámica a partir de rangos de direcciones predefinidas, o de manera estática en función de la dirección MAC del peticionario. La información es válida durante un período de tiempo dado (un contrato) que se puede renovar y configurar.

DHCP es un programa basado en **BOOTP** (*Bootstrap Protocol*). Cuando el cliente busca contactar un servidor, BOOTP le facilita la información de direccionamiento. DHCP gestiona las renovaciones. BOOTP se basa en el protocolo de transporte UDP.

Un cliente del servicio no tiene información de red disponible en el momento del inicio. Debe encontrar sólo un servidor DHCP. Para ello, BOOTP efectúa un broadcast mediante la IP 255.255.255.255 con una trama que contiene sus datos (como su dirección MAC) y los datos deseados (tipo de petición, aquí DHCPDISCOVER, puerto de conexión, etc.). Se manda el broadcast por definición a todos los anfitriones de la red local. Cuando el servidor DHCP detecta la trama, efectúa también un broadcast (el anfitrión cliente aún no tiene IP) con la información básica que desea el anfitrión (DHCPOFFER, primeros parámetros). El anfitrión establece una primera configuración; luego pide confirmación de la IP (DHCPREQUEST). El servidor DHCP confirma (DCHPHACK). El contrato está confirmado y el cliente dispone entonces de toda la información válida.

## 2. Servidor dhcpd

### a. Inicio

El servidor **dhcpd** es un servicio (daemon) iniciado mediante un script (/etc/init.d/dhcpd). Se configura con el fichero /etc/dhcpd.conf. Las direcciones IP asignadas se colocan en /var/lib/dhcp/dhcpd.leases.

```
# service dhcpcd start
```

O:

```
# /etc/init.d/dhcpcd start
```

## 3. Información básica

El fichero de configuración de un servidor es bastante simple si se limita a unos ajustes básicos.

```
ddns-update-style none; # ninguna actualización del DNS por DHCP
option domain-name "pepito.es"; # nombre de dominio transmitido al cliente
option domain-name-servers 192.168.1.254; # lista de los DNS separados
con comas
default-lease-time 21600; # duración del arrendamiento por defecto en segundos
sin petición explícita
max-lease-time 43200; # duración máx del arrendamiento si la petición del cliente
es más elevada
```

Como dhcpcd puede gestionar varias subredes, se debe especificar las reglas que hay que aplicar para cada subred. En general en el contexto de una pequeña red, sólo habrá un único bloque, pero todos los casos se pueden considerar. Si está seguro de que sólo tiene una red, puede omitir la declaración de la subnet (subred).

```
# Gestión de la subred 192.168.1.0
subnet 192.168.1.0 netmask 255.255.255.0
{
    option routers 192.168.1.254; # pasarela para esta red
    option subnet-mask 255.255.255.0; # máscara de subred
    range 192.168.1.2 192.168.1.250; # Configuración del rango DHCP

    # Caso de asignación de IP estáticas
    fuera puesto1
    {
        hardware ethernet 00:A0:ad:41:5c:b1; # Dirección MAC
        fixed-address 192.168.1.1; # esta máquina tendrá la IP 192.168.1.1
    }
}
```

Algunos clientes DHCP ignoran totalmente el hecho de que un servidor DHCP puede asignar un nombre (hostname) al anfitrión de manera dinámica. En el ejemplo anterior, la máquina con el IP 192.168.1.1 debería obtener el nombre puesto1. Ahí tiene un ejemplo:

```
# los clientes recibirán los nombres de los host declarados
use-host-decl-names on;
host puesto1
{
    hardware ethernet 00:A0:ad:41:5c:b1; # Dirección MAC
    fixed-address 192.168.1.1; # esta máquina tendrá la IP 192.168.1.1
y el nombre puesto1
}
```

También puede trabajar caso por caso:

```
host puesto2
{
    hardware ethernet 00:A0:ad:41:5c:b2; # Dirección MAC
    fixed-address 192.168.1.251; # este host tendrá la IP 192.168.1.251
    option host-name "puesto2"; # este host tendrá como nombre puesto2
}
```

## 4. Lado cliente

En Linux y las distribuciones de tipo Red Hat, Fedora, Mandriva, openSUSE, etc., modifique el fichero /etc/sysconfig/network-script/ifcfg-xxx especificando dhcp para **BOOTPROTO** y vuelva a iniciar la conexión de red (**ifdown**, luego **ifup**).

El cliente dhcpcd permite activar dhcp en una interfaz de red. Por ejemplo, el método más sencillo para eth0 es:

```
# dhcpcd eth0 &
```

Puede transmitir opciones a dhcpcd para que se encargue de varias posibilidades. Entre estas opciones tenemos:

- -D: autoriza la modificación del nombre de dominio;
- -H: autoriza la modificación del nombre de anfitrión;

- **-R**: evita la sobreescritura del fichero resolv.conf;
- **-l**: permite modificar el período de vigencia (leasetime) en segundos.

```
# dhcpcd -D -H -l 86400 eth0
```

# Servidor DNS

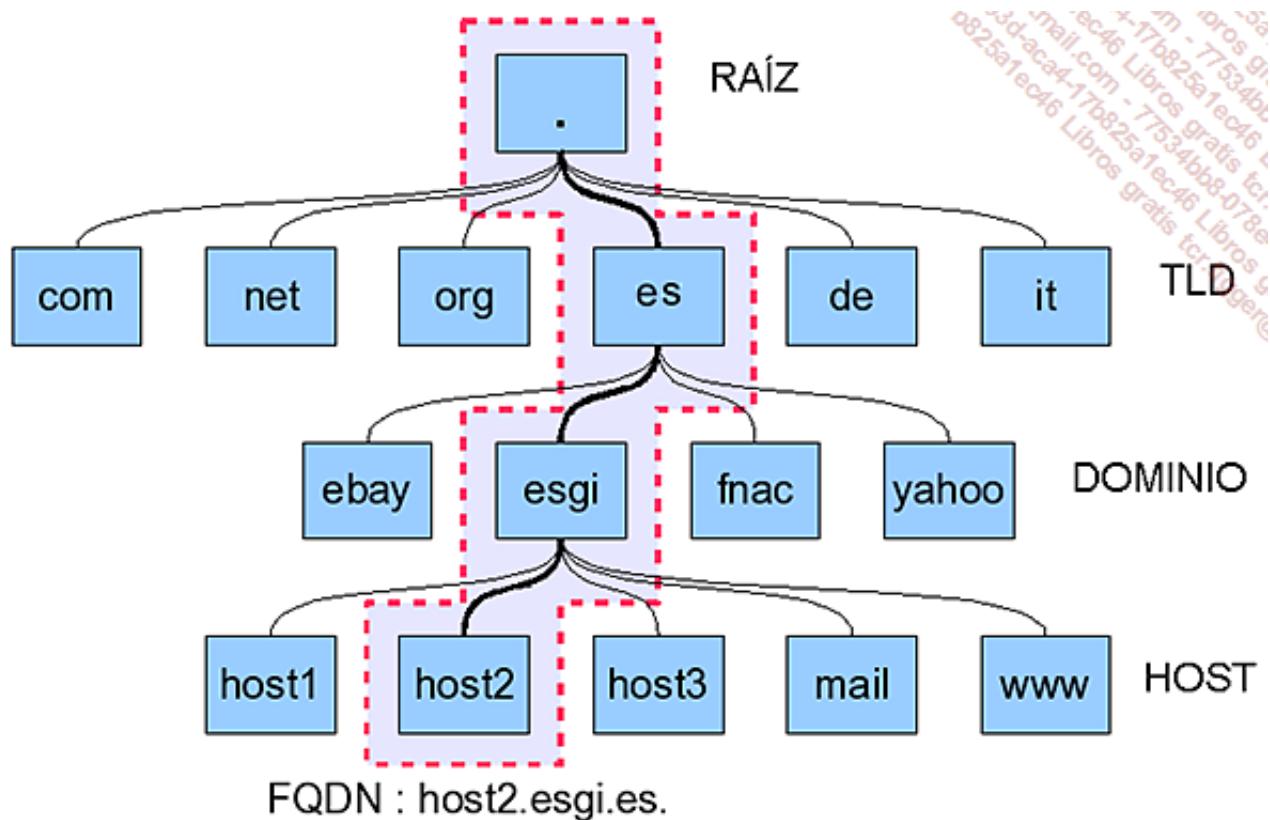
## 1. Presentación

El sistema de Nombres de Dominios **DNS** (*Domain Name System*) transforma los nombres de anfitrión en direcciones IP: es la **resolución de nombre** y transforma las direcciones IP en nombres de anfitrión: es la **resolución inversa**. Permite agrupar las máquinas por dominios de nombre. Facilita información de encaminamiento y correo electrónico.

El DNS permite referirse a sistemas basados en IP (los *anfitriones*) empleando nombres amenos (los *nombres de dominios*). El interés de un DNS es obvio. Los nombres de dominio son más fáciles de recordar y, si su dirección IP cambia, el usuario no se da cuenta. Por lo tanto, se entiende que el DNS es un servicio crítico para Internet.

Los nombres de dominio van separados por puntos. Como cada elemento se puede componer de 63 caracteres, sólo puede haber un máximo de 127 elementos y el nombre completo no debe superar 255 caracteres. El nombre completo no abreviado se llama **FQDN** (*Fully Qualified Domain Name*). En un FQDN, el elemento más a la derecha se llama **TLD** (*Top Level Domain*), el más a la izquierda representa al anfitrión y, por tanto, la dirección IP.

El DNS contiene una configuración especial para los enruteadores de correo electrónico (definiciones MX) que permiten una resolución inversa, un factor de prioridad y tolerancia a errores.



Una zona es una parte de un dominio gestionado por un servidor particular. Una zona puede gestionar uno o varios subdominios y se puede repartir un subdominio en varias zonas. Una zona representa una unidad de administración, de la cual puede ser responsable una persona.

## 2. Inicio

El servicio se llama **named**.

```
# service named start  
o:  
# /etc/init.d/named start
```

## 3. Configuración de Bind

**Bind** (*Berkeley Internet Name Daemon*) es el servidor de nombres más utilizado en Internet. Bind 9 soporta el IPv6, los nombres de dominio unicode, el multithread y muchas mejoras de seguridad.

### a. Configuración general

La configuración global de Bind se encuentra en el fichero `/etc/named.conf`. La configuración detallada de las zonas se coloca en `/var/lib/named`. `/etc/named.conf` se compone de dos partes. La primera corresponde a la configuración global de las opciones de Bind. La segunda es la declaración de las zonas para los dominios individuales. Los comentarios empiezan con una `#` o `//`.

 Cuidado: a veces ocurre (en particular en RHEL 4.x) que la configuración de Bind sea «chrootead» (movida a una estructura específica de la cual no puede salir el servicio, que tampoco puede acceder al resto de la estructura). En Centos y RHEL 4.x y superiores, `named.conf` está en `/var/named/chroot/etc/`. Se puede modificar este modo cambiando el fichero de configuración `/etc/sysconfig/named`.

```
# cat /etc/sysconfig/named  
...  
CHROOT=/var/named/chroot  
...
```

En este caso, todos los ficheros de configuración, incluidas las zonas, son relativos a esta ruta. Veamos un fichero `named.conf` básico.

```
options {  
    directory "/var/lib/named";  
    forwarders { 10.0.0.1; };  
    notify no;  
};  
zone "localhost" in {  
    type master;  
    file "localhost.zone";  
};  
zone "0.0.127.in-addr.arpa" in {  
    type master;  
    file "127.0.0.zone";  
};  
zone "." in {  
    type hint;  
    file "root.hint";  
};
```

### b. Sección global

La configuración global está ubicada en la sección **options**. Veamos un detalle de algunas opciones importantes (se debe especificar el punto y coma):

- **directory "filename";**: ubicación de los ficheros que contienen los datos de las zonas.
- **forwarders { direccion-ip; };**: si el propio servidor bind no puede resolver la petición, se la manda a un servidor DNS exterior, por ejemplo el del proveedor de acceso.
- **listen-on-port 53 {127.0.0.1; direccion-ip; };**: puerto de escucha del DNS, seguido de las direcciones de escucha. Aquí se indican las direcciones IP de las interfaces de red de la máquina. No hay que olvidar 127.0.0.1.
- **allow-query { 127.0.0.1; red; };**: máquina(s) o red(es) autorizadas a utilizar el servicio DNS. Por ejemplo, 192.168.1/24. Si la directiva no está, se autoriza todo.
- **allow-transfer { 192.168.1.2; };**: máquina(s) o red(es) autorizadas a copiar la base de datos en el caso de una relación maestro y esclavo. Por defecto, no se autoriza ninguna copia.
- **notify no;**: se notifica o no a los otros servidores DNS de un cambio en las zonas o de un nuevo arranque del servidor.

### c. Sección de zonas

Para cada dominio o subdominio, se definen dos secciones **zona**. La primera contiene los datos de resolución de nombre (Nombre hacia IP) y la segunda, los datos de resolución inversa (IP hacia Nombre). En cada caso, la zona puede ser maestro (**Master**) o esclavo (**Slave**):

- **Master**: el servidor contiene la totalidad de los registros de la zona en sus ficheros de zona. Cuando recibe una petición, busca en sus ficheros (o en su caché) la resolución.
- **Slave**: el servidor no contiene por defecto ningún registro. Se sincroniza con un servidor maestro del cual recupera toda la información de zona. Se puede colocar esta información en un fichero. En este caso, el esclavo almacena una copia local de la base de datos. Durante la sincronización, el número de serie de esta copia se compara con el del maestro. Si los números son diferentes, se hace una nueva copia; si no es el caso, se sigue utilizando la anterior.

### d. Zona de resolución

Se suele llamar **zona**. Para cada dominio o subdominio, indica en qué fichero se coloca la información de la zona (o sea, entre otras cosas, las direcciones IP asociadas a cada anfitrión), su tipo (maestro o esclavo), si se autoriza o no la notificación, la dirección IP del servidor DNS maestro en el caso de un esclavo, etc.

El nombre de la zona es muy importante, ya que determina el dominio de búsqueda. Cuando el DNS recibe una petición, busca una correspondencia en todas las zonas.

```
zona "dominio.org" {
    type    "master";
    file    "dominio.org.zona";
};
```

- **type**: master o slave;
- **file**: nombre del fichero que contiene los datos de la zona. No hay reglas precisas para atribuir nombres, pero por razones de legibilidad se aconseja darle el mismo nombre que la zona tanto para una zona maestra como para una esclava. Para un master, es el original completado por usted si es preciso. Para un esclavo no es obligatorio. Si está presente, será una copia del master, sincronizada.
- En el caso de un Master, se puede añadir **allow-transfer** (servidores autorizados a duplicar la zona)

y **notify yes** (indica una actualización o un nuevo arranque para los esclavos).

En caso de Slave: se añade la directiva **masters** para indicar a partir de qué servidor Master duplicar.

### e. Zona de resolución inversa

Para cada red o subred IP (o rango de direcciones) se define una zona de resolución inversa cuyo fichero contiene una asociación IP hacia el nombre de máquina. Resulta ser casi lo mismo que la zona de resolución, excepto que se debe respetar una convención de atribución de nombres:

- El nombre de la zona termina siempre por un dominio especial **.in-addr.arpa**.
- Primero se debe determinar qué red debe cubrir la zona (caso de las subredes). Para nosotros: una red de clase C 192.168.1.0, o sea **192.168.1/24**.
- Se invierte el orden de los bytes en la dirección: **1.168.192**.
- Se añade **in-addr.arpa**. Así, nuestro nombre de zona será **1.168.192.in-addr.arpa**.
- Para el resto se aplican las mismas observaciones que para la zona de resolución.

```
Zona "1.168.192.in-addr.arpa" {
    type    master;
    file    "192.168.1.zona";
};
```

### f. Ejemplo

Supongamos un dominio ([dominio.org](#)) en una red de clase C 192.168.1.0. Supongamos dos servidores DNS 192.168.1.1 Master y 192.168.1.2 Slave.

#### En el Master

```
zona "dominio.org" {
    type    master;
    file    "dominio.org.zona";
    allow-transfer { 192.168.1.2; } ;
    notify yes;
};
zona "1.168.192.in-addr.arpa" {
    type    master;
    file    "192.168.1.zona";
    allow-transfer { 192.168.1.2; } ;
    notify yes;
};
```

#### En el Slave

```
zona "dominio.org" {
    type    slave;
    file    "dominio.org.zona";
    masters   { 192.168.1.1; };
};
zona "1.168.192.in-addr.arpa" {
    type    slave;
    file    "192.168.1.zona";
    masters   { 192.168.1.1; };
};
```

---

## g. Zonas especiales

La zona raíz «.» permite especificar los servidores raíces. Cuando ninguna zona consigue resolver una petición, se utiliza la zona raíz por defecto y ésta reenvía a los servidores raíces.

La zona de loopback no es necesaria, aunque sí útil. Sirve de **caché DNS**. Cuando una petición llega al servidor y éste no posee la información de resolución, la va a pedir a los servidores DNS raíces, que bajarán la información. Se coloca entonces esta información en caché. ¡Por lo tanto, los accesos siguientes serán mucho más rápidos!

## 4. Ficheros de zonas

### a. Definiciones

Los ficheros de zonas utilizan varios términos, caracteres y abreviaciones específicas.

- **RR:** *Resource Record*. Nombre de un registro DNS (los datos del DNS).
- **SOA:** *Start Of Authority*. Permite describir la zona.
- **IN:** *the Internet*. Define una clase de registro que corresponde a los datos de Internet (IP). Si no se especifica para los registros, será la que haya por defecto.
- **A:** *Address*. Permite asociar una dirección IP a un nombre de anfitrión. Para Ipv6 es AAAA.
- **NS:** *Name Server*. Designa un servidor DNS de la zona.
- **MX:** *Mail eXchanger*. Designa un servidor de correo electrónico con un indicador de prioridad. Cuanto más bajo es el valor, más elevada es la prioridad.
- **CNAME:** *Canonical Name*. Permite añadir alias: relacionar un nombre a otro. Se pueden crear alias sobre nombres de anfitrión y también sobre alias.
- **PTR:** *Pointer*. En una zona de resolución inversa, hace que una IP apunte a un nombre de anfitrión.
- **TTL:** *Time To Live*. Duración de vida de los registros de la zona.
- @: en las declaraciones de la zona es un alias (carácter de sustitución) para el nombre de la zona declarada en /etc/named.conf. Así, si la zona se llama dominio.org, @ vale dominio.org. En la declaración del administrador de la SOA, sustituye de manera puntual el punto en la dirección de correo electrónico.
- El punto «.»: Si se olvida el punto al final de la declaración de anfitrión, el nombre de la zona está concatenado al final del nombre. Por ejemplo, para la zona dominio.org, si se escribe **puesto1**, esto equivale a **puesto1.dominio.org**. Si se escribe **puesto1.dominio.org** (sin el punto final), entonces se obtiene el resultado siguiente: **puesto1.dominio.org.dominio.org**. Para evitarlo, debe escribir **puesto1.dominio.org**. (observe el punto al final).
- Algunos registros necesitan una noción de duración, que se suele expresar en segundos, pero también a veces con abreviaciones:
  - **1M:** un minuto, o sea, 60 segundos (1M, 10M, 30M, etc.);
  - **1H:** una hora, 3.600 segundos;
  - **1D:** un día, 86.400 segundos;
  - **1W:** una semana, 604.800 segundos;

- **365D**: un año, 31.536.000 segundos.

 Cuidado, esto es muy importante: en los ficheros de zonas, NO SE DEBE NUNCA EMPEZAR UNA LÍNEA CON ESPACIOS O TABULACIONES. No funciona para nada: se interpretarían los espacios o tabulaciones como parte del nombre indicado, de la dirección o de la opción.

## b. Zona

Empiece primero por una directiva **TTL**, que indica el tiempo de vida de la zona en segundos. Esto significa que cada registro de la zona será válido durante el tiempo indicado por **TTL** (nota: es posible modificar este valor para cada registro). Durante este tiempo, los otros servidores de nombres remotos pueden colocar los datos en caché. Un valor elevado permite reducir el número de peticiones efectuadas y alargar los plazos entre las sincronizaciones.

\$TTL 86400

A continuación de las directivas TTL, registre el recurso **SOA**:

```
<domain>    IN      SOA      <primary-name-server> <hostmaster-email> (
                    <serial-number>
                    <time-to-refresh>
                    <time-to-retry>
                    <time-to-expire>
                    <minimum-TTL> )
```

**domain**: es el nombre de zona, el mismo nombre utilizado en el fichero /etc/named.conf. •Está permitido sustituirlo por una @ si no se hace así, de lo contrario, no se debe olvidar finalizarlo con un punto (para evitar la concatenación).

- **primary-name-server**: el nombre del servidor primario en esta zona. No olvide declararlo en la lista de hosts (registros PTR o A).
- **hostmaster-email**: dirección de correo electrónico del administrador del servidor de nombres. Dado que la arroba @ ya está reservada para otro uso, se utiliza un punto para reemplazarlo. Así pues, «admin@dominio.org» debería escribirse «**admin.dominio.org**».
- **serial-number**: es un número de serie que se debe incrementar manualmente a cada modificación del fichero de zonas con el objetivo de que el servidor de nombres sepa que debe volver a cargar esta zona. El número es utilizado en la sincronización con los servidores esclavos. Si el número de serie coincide con el de la última sincronización, los datos no necesitan ser actualizados. Por convención se usan diez cifras con el formato **YYYYMMDDNN** (año-mes-día-número).
- **time-to-refresh**: especifica a cualquier servidor esclavo cuánto tiempo debe esperar antes de consultar al servidor de nombres maestro si ha habido cambios en la zona.
- **time-to-retry**: indica al servidor esclavo cuánto tiempo debe esperar antes de emitir de nuevo una petición de actualización si el servidor maestro no ha respondido. La petición tendrá lugar cada time-to-retry segundos.
- **time-to-expire**: si, a pesar de las tentativas de contacto cada time-to-retry segundos, el servidor no ha respondido al final del período indicado por time-to-expire, el servidor esclavo cesa de responder a las peticiones para esta zona.
- **minimum-TTL**: el servidor de nombres solicita a los otros servidores de nombres poner en la caché la información para esta zona durante, al menos, el período indicado.

Diríjase a continuación a los registros **NS** (*Name Server*), donde especificará los servidores de nombres de

```

@ IN SOA dns1.dominio.org. hostmaster.dominio.org. (
    2005122701 ; serial
    21600       ; refresh de 6 horas
    3600        ; intentar cada 1 horas
    604800      ; tentativas expiran tras una semana
    86400       ; TTL mínimo de un día

```

esta zona.

```

IN NS dns1
IN NS dns2

```

 Cuando no se especifique al comienzo de la línea un nombre de anfitrión o de zona (completo o con arroba @), querrá decir que se utilizará el mismo que el de la línea superior. En tanto en cuanto no se precise uno nuevo, será el último indicado el que se va a utilizar. Según lo afirmado anteriormente, las líneas podrían ser las siguientes:

```

@ IN NS dns1
@ IN NS dns2

```

```

dominio.org.   IN   NS   dns1
dominio.org.   IN   NS   dns2

```

 Observe la ausencia del punto tras el nombre de host; por ello, dominio.org está concatenado para obtener dns1.dominio.org.

IN NS dns1

equivale a:

IN NS dns1.dominio.org.

Diríjase a continuación a la relación de servidores de correo electrónico de zona. El valor numérico que aparece tras la MX establece la prioridad. Cuanto más bajo sea este valor, más prioritario será el servidor y más susceptible de ser contactado en primer lugar. Si los valores son idénticos, el correo es redistribuido de forma homogénea entre los servidores. Si un servidor no responde (sobrecargado, averiado) la petición bascula automáticamente a otra máquina.

```

IN MX 10 mail
IN MX 15 mail2

```

que desea es que una máquina responda pasando por el FQDN dominio.org sin precisar el anfitrión (por ejemplo, solicitar <http://dominio.org> en lugar de <http://www.dominio.org>), puede, en este momento, declarar una dirección IP para dicho servidor. Así pues, el comando **ping dominio.org** responderá i192.168.1.3!

IN A 192.168.1.3

Puede declarar ahora otros anfitriones tales como servidores de nombres, de correo, de puestos, etc.

```

dns1      IN   A   192.168.1.1
dns2      IN   A   192.168.1.2
server1   IN   A   192.168.1.3
server2   IN   A   192.168.1.4
puesto1  IN   A   192.168.1.11

```

puesto2	IN	A	192.168.1.12
puesto3	IN	A	192.168.1.13

Vemos que no han sido declarados los servidores mail y mail2, y que tampoco se ha indicado el servidor Web y ftp. Vamos a utilizar alias que apunten estos nombres de host a otros.

mail	IN	CNAME	server1
mail2	IN	CNAME	server2
www	IN	CNAME	server1
ftp	IN	CNAME	server1

Ha finalizado la configuración de la zona; ahora hay que ocuparse de la zona de resolución inversa.

### c. Zona de resolución inversa

La zona de resolución inversa es casi idéntica a la anterior, sólo que los registros A son reemplazados por los registros PTR destinados a traducir una IP en un anfitrión. De ser posible, el TTL y la declaración SOA deben ser idénticos (excepto el nombre de zona). Además, debe poner los registros NS.

IN	NS	dns1.dominio.org.
IN	NS	dns2.dominio.org.

Usted no estará obligado a poner en la zona de resolución inversa la traducción de las direcciones IP del DNS, ya que es el propio DNS el que resuelve su propio nombre! Sin embargo, hacerlo puede acelerar el proceso, ya que el DNS no debe ejecutar una consulta a sí mismo. Diríjase a los registros PTR traduciendo la dirección IP de cada anfitrión.

1	IN	PTR	dns1.dominio.org.
2	IN	PTR	dns2.dominio.org.
3	IN	PTR	server1.dominio.org.
4	IN	PTR	server2.dominio.org.
11	IN	PTR	puesto1.dominio.org.
12	IN	PTR	puesto2.dominio.org.
13	IN	PTR	puesto3.dominio.org.

Teóricamente, una misma IP puede atribuirse a varios anfitriones; los RFC no son muy explícitos sobre esta posibilidad, lo que, al final, puede crear problemas.

## 5. Diagnóstico de los problemas de configuración

El comando **named-checkconf** comprueba la sintaxis del fichero **named.conf**. Debe facilitarle el fichero como parámetro. La salida indicará las líneas que dan problemas.

El comando **named-checkzone** comprueba la sintaxis de un fichero de zona (incluso de resolución inversa). Debe especificarle el nombre del fichero de zona como parámetro.

### a. Consulta dig y host

El programa **dig** es una herramienta de consulta avanzada de servidor de nombres capaz de restituir todos los datos de las zonas.

```
> dig tele2.es
; <>> DiG 9.4.1-P1 <>> tele2.es
```

```

;; global options: printcmd
;; Got answer:
;; ->>>HEADER<<- opcode: QUERY, status: NOERROR, id: 63972
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;tele2.es.           IN      A

;; ANSWER SECTION:
tele2.es.          86363   IN      A       212.27.48.10

;; Query time: 1 msec
;; SERVER: 10.23.254.240#53(10.23.254.240)
;; WHEN: Wed May 14 09:36:09 2008
;; MSG SIZE  rcvd: 41

```

Por defecto, dig sólo restituye la dirección del host que se ha pasado como parámetro. En caso de éxito, el estado vale **NOERROR**; **ANSWER** indica el número de respuestas y dicha respuesta se sitúa debajo de la sección **ANSWER**. Para obtener una resolución inversa existen dos soluciones.

```
$ dig 10.48.27.212.in-addr.arpa ptr
```

o más sencillamente:

```

$ dig -x 212.27.48.10

; <>> DiG 9.4.1-P1 <>> -x 212.27.48.10
;; global options: printcmd
;; Got answer:
;; ->>>HEADER<<- opcode: QUERY, status: NOERROR, id: 60222
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;10.48.27.212.in-addr.arpa.    IN      PTR

;; ANSWER SECTION:
10.48.27.212.in-addr.arpa. 86400 IN      PTR      www.tele2.es.

;; Query time: 31 msec
;; SERVER: 10.23.254.240#53(10.23.254.240)
;; WHEN: Wed May 14 09:36:51 2008
;; MSG SIZE  rcvd: 68

```

En la primera sintaxis, observe que puede añadir un parámetro de consulta. Los principales se presentan a continuación.

- **a**: únicamente la dirección;
- **any**: toda la información correspondiente al dominio;
- **mx**: los servidores de mensajería;
- **ns**: los servidores de nombres;
- **soa**: la zona Start of Authority;
- **hinfo**: información sobre el anfitrión;
- **txt**: texto de descripción;
- **ptr**: zona reversa del anfitrión;
- **axfr**: lista de todos los anfitriones de la zona.

```

$ dig tele2.es any
; <>> DiG 9.4.1-P1 <>> tele2.es any
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28893
;; flags: qr aa; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 8

;; QUESTION SECTION:
;tele2.es.                      IN      ANY

;; ANSWER SECTION:
tele2.es.          86400   IN      NS      freens2-g20.tele2.es.
tele2.es.          86400   IN      A       212.27.48.10
tele2.es.          86400   IN      NS      freens1-g20.tele2.es.
tele2.es.          86400   IN      MX     20 mx2.tele2.es.
tele2.es.          86400   IN      SOA    freens1-g20.tele2.es.
hostmaster.proxad.net. 2008051001 10800 3600 604800 86400
tele2.es.          86400   IN      MX     10 mx1.tele2.es.

;; ADDITIONAL SECTION:
freens2-g20.tele2.es. 86400   IN      A       212.27.60.20
mx1.tele2.es.         86400   IN      A       212.27.48.6
mx2.tele2.es.         86400   IN      A       212.27.42.56
freens1-g20.tele2.es. 86400   IN      A       212.27.60.19
mx2.tele2.es.         86400   IN      A       212.27.42.58
mx1.tele2.es.         86400   IN      A       212.27.48.7
mx2.tele2.es.         86400   IN      A       212.27.42.57
mx1.tele2.es.         86400   IN      A       212.27.42.59

;; Query time: 9 msec
;; SERVER: 10.23.254.240#53(10.23.254.240)
;; WHEN: Wed May 14 09:35:32 2008
;; MSG SIZE  rcvd: 318

```

La herramienta **host** devuelve el mismo resultado, pero quizá de manera más sencilla.

```

$ host tele2.es
tele2.es has address 212.27.48.10
tele2.es mail is handled by 10 mx1.tele2.es.

$ host -t any tele2.es
tele2.es has address 212.27.48.10
tele2.es name server freens1-g20.tele2.es.
tele2.es has SOA record freens1-g20.tele2.es. hostmaster.proxad.net.
2008051001 10800 3600 604800 86400
tele2.es mail is handled by 10 mx1.tele2.es.

$ host -a tele2.es
Trying "tele2.es"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64513
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 2

;; QUESTION SECTION:
;tele2.es.                      IN      ANY

;; ANSWER SECTION:
tele2.es.          86140   IN      A       212.27.48.10
tele2.es.          86140   IN      NS      freens1-g20.tele2.es.
tele2.es.          86140   IN      SOA    freens1-g20.tele2.es.
hostmaster.proxad.net. 2008051001 10800 3600 604800 86400
tele2.es.          86140   IN      MX     10 mx1.tele2.es.

```

; ; ADDITIONAL SECTION:  
freens1-g20.tele2.es. 86140 IN A 212.27.60.19  
mx1.tele2.es. 86140 IN A 212.27.48.7

Received 176 bytes from 10.23.254.240#53 in 4 ms

# Correo electrónico

## 1. Fundamentos

- Cuando un cliente (un usuario) manda un mensaje, utiliza un **MUA** (*Mail User Agent*), por ejemplo Outlook Express, Thunderbird, Evolution, Kmail, Mutt, etc.
- El **MUA** envía el mensaje al **MTA** (*Mail Transport Agent*). El MTA estudia la dirección electrónica para aislar al usuario y dominio de destino. Luego verifica la información DNS de tipo **MX** (*Mail eXchanger*) para el dominio elegido, para saber a qué servidor transmitir el correo. Si ningún MTA está disponible, se coloca el mensaje en fila de espera y se reenvía más tarde (el plazo depende de la configuración del MTA).
- El MX puede ser o bien otro MTA, que tendrá el papel de enrutador (en caso de una redirección hacia un subdominio, por ejemplo) o bien un **MDA** (*Mail Delivery Agent*). El MDA coloca el mensaje en un fichero temporal, puede filtrarlo, etc.
- En este nivel, el destinatario recibe el mensaje: o lo recupera al leer directamente el fichero temporal (caso del comando mail, por ejemplo), o pasa por un protocolo de tipo **POP** o **IMAP**.
- El protocolo de transporte de mensajes es el **SMTP** (*Simple Mail Transfer Protocol*) en el puerto 25.
- Los protocolos de recepción de mensajes son o **POP** (*Post Office Protocol*) en el puerto 110 (POP3), o **IMAP** (*Internet Message Access Protocol*).

Dos suites de correo electrónico comparten la parte esencial del mercado en Unix: **sendmail** y **postfix**.

La serie libre **sendmail** es la más conocida y más utilizada. Eric Allman creó Sendmail en 1981, que fue integrado en BSD 4.2 en 1983. En 2000 se estimaba que lo usaban a más de 100 millones de servidores de correo electrónico. Mientras no haya que modificar su configuración básica, sendmail es idóneo. Si desea ir más lejos, le hará falta comprarse un libro completo sobre este tema. La configuración de sendmail es tan compleja que se inventó un lenguaje de macros llamado **m4** sólo para él. Así no se edita (o poco) el fichero de configuración de sendmail: se edita el fichero fuente de las macros y se vuelve a compilar: m4 va a crear el fichero de configuración de sendmail. Sendmail se ha convertido en un monstruo en lo que respecta a potencia y configuración.

El producto **postfix** tiende a ser cada vez más utilizado no tanto por aquellos a quienes sendmail ha decepcionado, sino por los que temen no saber configurarlo. Es una alternativa a sendmail. Los objetivos de sus desarrolladores (entre los cuales algunos son los de sendmail) son:

- la compatibilidad con sendmail;
- la rapidez (más de un millón de mensajes al día en un simple Pentium 4);
- la sencillez de administración (fichero de configuración simple y legible);
- la seguridad (se puede chrootear);
- la modularidad (descomposición de los tratamientos).

Nos quedaremos con la sencillez. En efecto, se puede configurar postfix con un solo comando sin necesidad de editar un fichero. Vamos a utilizar este servidor.

## 2. postfix

### a. Configuración sencilla

La configuración de **postfix** se encuentra en /etc/postfix/main.cf. Se modifican sus valores o bien manualmente, o con la ayuda del comando **postconf**.

Postfix inicia primero un servicio maestro, **master**, encargado de los procesos secundarios **smtpd,pickup** y **nqmgr**.

---

- En algunas distribuciones, hay que modificar la configuración por defecto que utiliza la serie sendmail. Por ejemplo, en Red Hat usted debe indicar que se utilice postfix en lugar de sendmail con el comando **alternatives**.
- 

```
#alternatives --set mta /usr/sbin/sendmail.postfix
```

Aplique una configuración básica con el comando **postconf**.

#### Dominio de origen de los mensajes

```
#postconf -e "myorigin = midominio.org"
```

#### De qué dominio recibir el correo

```
#postconf -e "mydestination = midominio.org"
```

#### De qué clientes transmitir el correo

```
#postconf -e "mynetworks = 192.168.1.0/24, 127.0.0.1"
```

#### En qué interfaces escuchar

```
#postconf -e "inet_interface = all"
```

Inicie el servicio.

```
#service postfix start
```

O:

```
#/etc/init.d/postfix start
```

## b. Alias de usuarios

Es posible ubicar alias para los usuarios locales en el fichero **/etc/aliases**. Por ejemplo, si los mensajes de webmaster, admin y root se deben redireccionar hacia manuel:

```
manuel: webmaster, admin, root
```

## c. Probándolo

El registro se ubica en /var/log/maillog. Pruebe el servidor de la manera siguiente (por ejemplo):

```
mail -s `echo $USER` root@server1 < /etc/passwd
```

Si todo funciona, obtendrá trazas:

```
Fri 26 11:38:18 estacion1 postfix/pickup[12357] : F145040154: uid=0
from <root>
```

```
Fri 26 11:38:18 estacion1 postfix/cleanup[12318] : F145040154:  
message-id=<20060126113017.F145040154:@estacion1.midominio.org>  
Fri 26 11:38:26 estacion1 postfix/nqmgr[3469] : F145040154:  
from=<root@estacion1.example.com>, size=314, nrcpt=1 (queue active)  
Fri 26 11:38:32 estacion1 postfix/smtp[12468] : F145040154:  
to=<root@server1>, relay=server1.midominio.org[192.168.1.1],  
delay=17, status=sent (250 ok dirdel)
```

### 3. POP e IMAP

Existen varias suites para gestionar POP e IMAP. Una suite se llama **cyrus-imap** y en principio está reservada a las grandes estructuras y a los servidores 100% dedicados al correo, o sea, donde los usuarios no se conectan.

Una solución se llama **dovecot**. Tras su instalación, basta con iniciarla como servicio para que todo funcione, o casi todo.

Edite el fichero `/etc/dovecot.conf` para comprobar los protocolos soportados.

```
protocols = imap pop3
```

Inicie el servicio:

```
service dovecot start
```

o:

```
/etc/init.d/dovecot start
```

Pruebe mandando un mensaje. Configure un cliente de mensajería para comprobar si el servidor POP funciona:

```
# telnet localhost 110  
trying 127.0.0.1...  
Connected to localhost.localdomain.  
Escape character is '^].'  
+OK POP3 localhost.localdomain server ready  
USER manuel  
+OK user name accepted, password please  
PASS password  
+OK Mailbox open, 1 messages  
STAT  
+OK 1 384  
TOP 1 99999  
<mensaje aquí>  
...  
DELE 1  
+OK Message deleted  
QUIT  
+OK bye
```

# Servicio HTTP Apache

## 1. Presentación

**Apache 2** es el servidor HTTP más utilizado actualmente en los servidores Web. Su configuración y su flexibilidad hacen que sea un servidor ineludible.

Cuando un servidor Apache recibe peticiones, puede distribuirlas a procesos hijos. La configuración permite iniciar procesos de manera anticipada y adaptar de manera dinámica este número según la carga.

Apache es modular. Cada módulo permite añadir funcionalidades al servidor. El módulo más famoso es probablemente el que gestiona el lenguaje PHP, «mod\_php». Cada módulo se añade mediante los ficheros de configuración y no hace falta iniciar de nuevo el servidor Apache: sólo se le da la orden de volver a leer su configuración.

Apache puede gestionar varios sitios Web al mismo tiempo, cada uno con su nombre, mediante los anfitriones virtuales.

## 2. Parada/Reinicio

El nombre del servicio depende de la distribución. A menudo se denomina **apache** o **httpd**. Según la distribución, inicie el servicio mediante el comando **service** o directamente por su nombre,/etc/init.d/apache.

- **/etc/init.d/httpd start:** se inicia;
- **/etc/init.d/httpd stop:** se para;
- **/etc/init.d/httpd restart:** se reinicia;
- **/etc/init.d/httpd reload:** pide a Apache que vuelva a leer su configuración sin reiniciarse.

Apache va acompañado de la herramienta **apachectl**, que retoma los parámetros (lista no exhaustiva) start, stop, status, reload, y sobre todo **configtest**, que valida o no el contenido del fichero de configuración de Apache.

## 3. Configuración

Se almacena la configuración principal en /etc/httpd/conf/httpd.conf: controla los parámetros generales del servidor Web, los anfitriones virtuales y los accesos. La configuración de los diferentes módulos está ubicada en /etc/httpd/conf.d. Los módulos están presentes en /etc/httpd/modules/. Por defecto la raíz del servidor, donde se colocan las páginas del sitio, está en /var/www o /srv/www. Esta posición depende de la directiva **DocumentRoot** en los ficheros de configuración.

## 4. Directivas generales

No es posible listar todas las directivas del fichero httpd.conf, pero algunas son importantes.

- **ServerRoot:** directorio que contiene los ficheros del servidor (configuración y módulos). Suele ser /etc/httpd.
- **Listen:** puertos en los cuales escucha el servidor Apache. Por defecto 80 (443 en https). Se pueden especificar varios puertos con varias directivas Listen. Si el servidor dispone de varias direcciones IP,

se puede añadir la IP al puerto asociado: **Listen 192.168.1.3:80**.

- **User:** usuario de los procesos Apache. Nunca se utiliza root, sino una cuenta creada para la ocasión, en general Apache.
- **Group:** ídem, pero para el grupo.
- **ServerAdmin:** dirección de correo electrónico del administrador.
- **ServerName:** nombre de anfitrión (y puerto) del servidor. No corresponde obligatoriamente al nombre de anfitrión de la máquina. En cambio, es válido. **ServerName www.midominio.org**
- **UseCanonicalName:** si vale **on**, Apache va a responder utilizando la información de ServerName y Port, y no la información mandada por el cliente. Por ejemplo, un http://192.168.1.3 se transforma en http://www.midominio.org.
- **UserDir:** nombre de un subdirectorio donde cada usuario puede ubicar sus ficheros HTML personales. En general, public\_html. Se accede a él con: **http://www.midominio.org/~login/page.html**
- **ErrorLog:** fichero donde se colocan los logs de error del servidor. /var/log/httpd/error\_log.
- **CustomLog:** registro de Apache. /var/log/httpd/access\_log.
- **Timeout:** período durante el cual el servidor espera emisiones/recepciones en el transcurso de una comunicación. Por omisión, 300 segundos.
- **KeepAlive:** define si el servidor puede ejecutar más de una petición por conexión. Está en off por defecto, pero si se pasa a on, Apache puede generar rápidamente unos procesos hijos para aligerar el exceso de carga.
- **MaxKeepAliveRequests:** número máximo de peticiones por conexión persistente. Un valor elevado puede aumentar el rendimiento del servidor. 100 por defecto.
- **KeepAliveTimeout:** período durante el cual el servidor (en general un proceso hijo) espera después de haber servido una petición. Por defecto, 15 segundos. Después de 15 segundos, se recibirá la petición por el servidor con un Timeout.
- **StartServers:** número de servidores creados en el momento del inicio. Por defecto, 8. Ya que Apache gestiona de manera dinámica el número de servidores hijos, este parámetro tiene poca importancia porque el número va a bajar o a aumentar rápidamente.

## 5. Gestión del rendimiento

- **MaxRequestPerChild:** número de peticiones que se pueden ejecutar con un proceso hijo antes de pararse. Por defecto, 4000. Eso permite una ocupación de la memoria más reducida, liberándola más rápidamente.
- **MaxClients:** límite del número total de peticiones que se pueden tratar simultáneamente. Por defecto, 150. El valor limita el riesgo de saturación del servidor.
- **MinSpareServers / MaxSpareServers:** según la carga de la máquina, Apache puede iniciar otros procesos servidores para adaptarse a la carga actual. Por defecto, son de 5 y 20. Estos dos valores determinan los números límite autorizados. Si un servidor está poco cargado con 15 procesos, Apache suprimirá algunos, pero guardará siempre al menos 5. Si el servidor está cargado con 10 procesos, Apache creará unos adicionales, hasta 20, para situaciones de alta «concurrency».
- **MinSpareThreads / MaxSpareThreads:** cada servidor hijo puede aceptar un determinado número de peticiones simultáneamente. Por eso utiliza los threads. Se fijan estos dos valores por defecto a 20 y 75. El mecanismo funciona como anteriormente.

- **ThreadsPerChild**: número de threads por defecto en el momento del inicio de un servidor hijo. Por defecto, 25.

## 6. Los directorios, alias y ubicaciones

### a. Directory

Las etiquetas **<Directory ruta>** y **</Directory>** permiten agrupar directivas que no sólo se aplicarán a la ruta (y a sus subdirectorios) de los datos. Se aconseja encarecidamente la directiva **Options**.

```
<Directory /var/www/html/images>
    Options +Indexes +FollowSymLinks
    DirectoryIndex index.php index.html
    Order allow, deny
    Allow from All
</Directory>

<Directory /var/www/html/cgi-bin>
    Options +ExecCGI
</Directory>
```

La directiva **Options** acepta los valores siguientes precedidos de + o - y separados por espacios:

- **All**: todas las opciones salvo MultiViews;
- **Indexes**: si acaso el directorio no contiene ficheros HTML por defecto (vea **DirectoryIndex**), se muestra el contenido del directorio en forma de listing;
- **ExecCGI**: se autoriza la ejecución de scripts CGI;
- **FollowSymLinks**: el servidor sigue los vínculos simbólicos.

La directiva **DirectoryIndex** especifica los ficheros html o cgi por defecto durante la carga de una URL.

**DirectoryIndex index.php index.html**

Durante la carga, sin precisar el nombre del fichero html, el servidor intentará cargar index.php. Si no está, entonces index.html. En el caso contrario, es la opción **Indexes** la que determina si el contenido debe mostrarse en forma de directorio.

La directiva **Allow** indica qué clientes tienen autorización para acceder al directorio. Puede ser **all**, un dominio, una IP, una IP truncada (subred), un par red/subred, etc. La directiva **Deny** prohíbe el acceso y se utiliza de la misma manera. Se determina el orden por la directiva **Order**.

### b. Alias

La directiva **Alias** permite crear un atajo entre el árbol lógico del sitio Web y una ruta del sistema de ficheros.

**Alias /help "/usr/share/doc/html"**

En este caso, la URL <http://www.misitio.org/help> no buscará en el directorio /var/www/html/help, sino en /usr/share/doc/html.

A diferencia de las etiquetas **<Directory>**, las etiquetas **<Location>** y **</Location>** permiten aplicar directivas basadas en la URL (y no los directorios).

```
<Location /help>
  Options +All -FollowSymLinks
  Order deny, allow
  Deny from all
  Allow from .midominio.org
</Location>
```

## 7. Anfitriones virtuales

Un servidor Apache es capaz de gestionar varios sitios Web en un mismo servidor. Existen varios métodos. El primero se basa en los nombres (varios sitios Web para un servidor); el otro, en las direcciones ip (una dirección IP para cada sitio Web). Vamos a considerar la primera versión.

La directiva **NameVirtualHost** especifica la dirección IP en la cual el servidor va a recibir las peticiones de acceso a los anfitriones virtuales.

Las etiquetas **<VirtualHost>** y **</Virtualhost>** permiten definir un anfitrío virtual.

```
NameVirtualHost 192.168.1.3

<VirtualHost 192.168.1.3>
  ServerName      www2.midominio.org
  ServerAdmin     webmaster@www2.midominio.org
  DocumentRoot   /var/www/www2.midominio.org/
  ErrorLog        logs/www2_error_log
  CustomLog       logs/www2_access_log
</VirtualHost>
```

Vuelva a iniciar Apache. Con un navegador (p. ej.: Firefox) se verifica si nuestro anfitrío virtual contesta con la URL <http://www2.midominio.org> (esta dirección debe declararse en /etc/hosts o bien ser conocida por el servidor de nombres y apuntar al servidor correcto).

Observe, sin embargo, que si usted pasa por <http://www.midominio.org> iya no obtiene el sitio por defecto! En efecto, cuando se declaran hosts virtuales, el primero de la lista se convierte en el anfitrío por defecto y prioritario.

- 
- Cuando se accede a un servidor, Apache busca primero una correspondencia entre el nombre de anfitrío especificado por la URL y cada Servername de los anfitriones virtuales. Si no se encuentra ninguna correspondencia exacta, se elige por defecto el primer anfitrío virtual haciendo abstracción de los parámetros globales.
- 

Añada un anfitrío virtual para el sitio principal.

```
<VirtualHost 192.168.1.3>
  ServerName      www.midominio.org
  ServerAdmin     webmaster@www.midominio.org
  DocumentRoot   /var/www/html
  ErrorLog        logs/error_log
  CustomLog       logs/access_log
</VirtualHost>
```

Cuidado, la regla anterior se aplica también con un nombre corto. Si usted inscribe <http://www> o <http://www2> se topará siempre con el anfitrío virtual por defecto. Hay que requerir <http://www.midominio.org> y <http://www2.midominio.org>.

Se pueden colocar en un anfitrío virtual todas las directivas deseadas (o casi).

# Requisitos y objetivos

## 1. Requisitos

- Disponer de los conocimientos sobre redes explicados en el capítulo La red.
- Disponer de los conocimientos de administración explicados en el capítulo Las tareas administrativas.
- Tener acceso como root.
- Estar conectado a una red.
- Tener la posibilidad de probar los accesos desde otra máquina de la red.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Manejar/trabajar con las bases de seguridad.
- Controlar los permisos SUID/SGID.
- Comprobar la integridad de un sistema de paquetes.
- Modificar la política de las contraseñas.
- Gestionar las conexiones y límites de los usuarios.
- Probar las contraseñas y la presencia de rootkits.
- Buscar y erradicar virus.
- Recibir boletines de seguridad.
- Efectuar actualizaciones de seguridad.
- Controlar la seguridad de la red con nmap.
- Parar los servicios inútiles.
- Proporcionar mayor seguridad a los servicios con los tcp wrappers.
- Configurar un firewall básico con Netfilter.

# Bases de seguridad

## 1. Seguridad informática

Los principales objetivos de la seguridad informática conciernen a:

- **La seguridad de la conexión:** se trata de controlar que los usuarios que se conectan dispongan efectivamente de la autorización para ello y de prohibirles el acceso al sistema en caso contrario.
- **La integridad de los datos:** se trata de conseguir que los ficheros y las bases de datos no estén corruptas y de mantener la coherencia entre los datos.
- **La confidencialidad de los datos:** el acceso a los datos para consulta y modificación se debe limitar únicamente a usuarios autorizados.

Usted dispone de varios medios:

- La autentificación de los usuarios mediante contraseña.
- Encriptar los datos.
- La seguridad física controlando el acceso de las personas a las salas informáticas, mediante circuitos físicos inviolables.
- La información relativa a los riesgos penales en los que se incurre en caso de infracción. Un «atraco» informático es un delito, no un juego.
- El control frecuente de los derechos de acceso a los ficheros y bases de datos.
- El control de los «checksum» de los ficheros para asegurar su integridad.
- La copia de seguridad regular de los datos.
- El control de los principales eventos del sistema.
- La instalación de cortafuegos (firewall) que controlan los accesos al sistema informático desde el exterior y evitan que los usuarios accedan a servicios externos sin querer o sin necesitarlo y, de esa manera, limitar el riesgo de propagación de virus.
- La instalación de un antivirus, incluso en Linux, si el servidor trata datos desde y hacia sistemas operativos susceptibles de tener virus.
- La instalación de herramientas antispams y antispywares, según el mismo principio, con el fin de evitar una intrusión y la saturación de los servidores de correo electrónico.
- Iniciar únicamente los servicios realmente útiles en el servidor y el cliente.

Algunos métodos sencillos permiten limitar los riesgos:

- Es posible definir un valor de umask restrictivo (p. ej.: 077) para extender a continuación los permisos de acceso de algunos ficheros.
- No puede dejar su terminal sin desconectarse o sin bloquearla.
- Hay que prestar atención a las fechas de última conexión logradas e infructuosas que se muestran en cada conexión.
- No permitir nunca el acceso, incluso en modo de sólo lectura, al fichero .profile.
- Nunca poner el . en primera posición del PATH, y controlar sus rutas.

## 2. Controlar los privilegios especiales

Los privilegios especiales de ejecución (bits SUID y SGID) suelen ser causa de inseguridad en el sistema. En efecto, un usuario malintencionado, aprovechando la falta de atención o la ausencia de un compañero o un administrador que no está desconectado de su consola, puede modificar los permisos de ciertos comandos a su favor. El ejemplo más habitual es el de reescribir un shell como un programa poco usado (por ejemplo sx) y darle los privilegios SUID. Al iniciar este comando, se puede convertir en root.

### Obtener el permiso de listar todos los ficheros

```
# chmod u+s cat
```

### Obtener un shell root

```
# cp /bin/sh /bin/sx
# chmod u+s /bin/sx
...
$ sx
# ...
```

comando siguiente permite buscar todos los ficheros que disponen de los bits SUID o SGID:

```
# find / -type f \(\ -perm -4000 -o -perm -2000 \)
# find / -type f \(\ -perm -4000 -o -perm -2000 \)
/bin/su
/bin/umount
/bin/eject
/bin/mount
/bin/ping
/bin/ping6
/sbin/unix2_chkpwd
/sbin/unix_chkpwd
/usr/bin/expiry
/usr/bin/write
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/gnotski
/usr/bin/mahjongg
/usr/bin/chfn
/usr/bin/yset
/usr/bin/wall
/usr/bin/crontab
/usr/bin/v4l-conf
/usr/bin/gnomine
/usr/bin/same-gnome
/usr/bin/gnotravex
/usr/bin/gnobot2
...
```

En la lista anterior, hay un intruso: `/usr/bin/yset`, que permite modificar las configuraciones de un servidor de sonido y que no necesita para nada disponer del privilegio SUID. Hay un problema.

```
# ls -l /usr/bin/yset
-rwsr-sr-x 1 root root 604040 may 19 21:28 /usr/bin/yset
# md5sum /usr/bin/yset
04ff72010ff1cf1c14d7706159cdf8bf  /usr/bin/yset
# ls -l /bin/bash
-rwxr-xr-x 1 root root 604040 sep 22 2007 /bin/bash
# md5sum /bin/bash
04ff72010ff1cf1c14d7706159cdf8bf  /bin/bash
```

Alguién ha vuelto a escribir un shell con otro nombre.

### 3. Comprobar los paquetes

El capítulo Instalación de Linux y de los programas ha tratado de toda la gestión de los paquetes de software. Entre las diversas opciones, algunas permiten controlar la autenticidad de un paquete. El sistema de paquetería RPM contiene, además del nombre del fichero, su tipo (configuración, binario, etc.) y en algunos casos (binario) la suma de control (checksum) MD5 del fichero.

Según el ejemplo anterior, ¿cómo restaurar el fichero yset? En tres etapas:

- Encontrar el paquete de origen:

```
# rpm -qf /usr/bin/yset  
yiff-2.14.5-0.pm.1
```

- Controlar el estado del paquete instalado:

```
# rpm -V yiff  
SM5....T    /usr/bin/yset
```

- S: el tamaño no es el correcto
- M: se han modificado los permisos
- 5: la suma de control MD5 es diferente
- T: la fecha de modificación no es la correcta.

- Vuela a instalar el paquete de origen según las modalidades propias de su distribución.

### 4. Política de la contraseña

Las contraseñas son la base de la autenticación de un usuario. Deben ser seguras. Sin embargo, suelen ser una fuerte laguna, tanto en el trabajo como en casa, e incluso en Internet:

- contraseña escrita en un post-it;
- empleo de un gestor de contraseña automático, él mismo sin contraseña;
- misma contraseña para todos los sitios Web y software;
- contraseña nunca cambiada;
- contraseña o cuenta comunes a toda la familia/servicio;
- contraseña bastante simple;
- etc.

No sirve de nada caer en la paranoia. Tiene que encontrar un punto medio. Si pide a los usuarios que modifiquen su contraseña demasiado a menudo, o si éste es demasiado difícil, tienden a apuntarla. Si es demasiado fácil y usted deja pasar demasiado tiempo, ello no es seguro.

Los usuarios deben elegir una buena contraseña, evitando la sencillez o más bien lo evidente: nombres de los hijos, de la esposa, de lugares, fecha de nacimiento y, en general, todo lo que importa y que es conocido del entorno profesional o personal.

Un punto medio puede ser modificar las reglas de cambio de contraseña con chage (o passwd), de forma que se establezca una duración para la validez de la contraseña de 40 días. Se han presentado los comandos de modificación de la política de gestión de las contraseñas en el capítulo Tareas administrativas - Administración de los usuarios.

```
# chage -l bean
Minimum:          7
Maximum:         40
Warning:          10
Inactive:         5
Last Change:      abr 10, 2008
Password Expires: may 20, 2008
Password Inactive: may 25, 2008
Account Expires:  ene 01, 2010
```

Los módulos PAM influyen en la política de gestión de las contraseñas, obligando en algunos casos a elegir uno más o menos complejo. Aunque parezca una paradoja, una contraseña debe ser fácil de recordar por un usuario, lo que no implica forzosamente que sea fácil de piratear (por John the Ripper, por ejemplo). Existen contraseñas que se pueden recordar por medios mnemotécnicos. También puede generar contraseñas de manera automática con la herramienta **pwgen**.

```
$ pwgen
uash6She lohJo7ae 0hphab3i ouRik9ie uM4va3im Neer7Eit eib3Hauy xo9Iuy5p
ahSiW0uf AhG6wail Yai6neeh phae4ioV deeL3aip Uz5ahzaa aiV5phee Aegaiy7x
ioPh1ahn Ong6Baib Eish4rip eik9Giel jen3Iepe xohduj7U aiP2keov So5ovaht
Voh9oxoe ahs2Meeg Ooch5xix Phe3yiuz eeCa5ohv aig9Ai3o Go4Ateeh Hee6thei
Rai6Daeh aid8ieNg Thah6ien daphaiG0 Iefai5oh Pheife6i Poora8ah Coh5Aida
ViC7ieth hohG5sei Aa9Jeilu eopoX8Si jooh3Eif dooPhail chohqu1G ieNgaе3o
wiCeisI3 aej6Piev eoThalFu ier2yeeb Eireili6 saiGhie2 XohRoo1a cahb2Yah
Guungah0 ube3vo0D oshol30p Pui6agh5 Ao7baeN1 foTek9Ei aeM3lala Ene2baol
geloV9ai Weeyu2ie Uvae2Vie dei0euL7 Xee9uaza ed8Eeghu eebiu2Ka zey0Liuh
be6Ailo1 eiph80hb Yahpahr4 aij4dahG oQu2chae Fe5eeg9c Hoosh6oh Iip8eiwe
AuPie0um Ahxai9eo Dae5oquu Ie7Viek8 pa2aew8B fohham7A fah10ogi ieH9vee8
saeC8sha Aejeey6i Eithowl yi9vei0L ohC7eegh IaTh4ohn ti6Fooke 0iche7oh
Tah9uos7 Paej2Iec chuid8ei aicoGh5l saiKeiw2 mae9mieY Ais9oanu Mah9xej3
Zi2nacai gaiM4thi sapalFah kie8oZo7 Po5uuho8 thae3Aim Ohjahgh9 Weike8ra
Cah4weiZ teoji00o vi0hei60 Zieha3ai Keip2bie bahR7bah ahSai0Ei afoh3Thi
eeNieTh8 Zei7eth8 uV5eichi kue1Eedi sueThe0V wohChe2u Ohl1zicu Loolsoo3
yahb9uSi EelieGh1 aeMiThi1 Oofoh8wu Ieyei5ka Roph7ape uem5quuK ahQu7eec
NahSha6A kooMou0y gulchaiJ hae2ku0Z uc3oeNgo xuSha7qu Iucai0fu uK4icewe
eP7aetig ahYai0ee Eetahfu8 yeep0oPi Veimaij3 Oht0aiPh buTh9oob ood4nieC
sah7Ahj1 koozah0J Vieb9Bit eeP9neel ealSohCe Afei4ohS eikahk0W rachoG4c
```

Estas contraseñas son seudoaleatorias. Si usted habla inglés (y geek/leet), estas contraseñas representan una pronunciación. Por ejemplo:

dooPhail: Do you fail?

Puede pedir que se generen contraseñas totalmente aleatorias con una longitud dada, en este caso de 10 caracteres:

```
$ pwgen -s -1 10
ER9BAgHsZH
```

## 5. Prohibir las conexiones

### a. /bin/false

Algunas cuentas no deben ser interactivas: se deben prohibir las conexiones desde una consola. Se pueden asignar estas cuentas a una aplicación, a un servicio, a una conexión FTP, etc., pero se debería rechazar la conexión: `no shell`!

En la lista de los shells autorizados, uno llama la atención:

```
$ cat /etc/shells
/bin/ash
/bin/bash
/bin/bash1
/bin/csh
/bin/false
/bin/ksh
/bin/sh
/bin/tcsh
/bin/true
/bin/zsh
/usr/bin/csh
/usr/bin/ksh
/usr/bin/passwd
/usr/bin/bash
/usr/bin/tcsh
/usr/bin/zsh
```

shell `/bin/false` no es realmente un shell. Prohíbe las conexiones interactivas. Ya hemos encontrado este comando `false` en el capítulo El shell y los comandos GNU: devuelve siempre falso. En cuanto `login` intenta ejecutar el shell de conexión, se rechaza al usuario.

```
$ cat /etc/passwd|grep false
avahi:x:104:106:User for Avahi:/var/run/avahi-daemon:/bin/false
haldaemon:x:101:102:User for haldaemon:/var/run/hal:/bin/false
icecream:x:102:103:Icecream Daemon:/var/cache/icecream:/bin/false
mail:x:8:12:Mailer daemon:/var/spool/clientmqueue:/bin/false
messagebus:x:100:101:User for D-Bus:/var/run/dbus:/bin/false
ntp:x:74:104:NTP daemon:/var/lib/ntp:/bin/false
polkituser:x:103:105:PolicyKit:/var/run/PolicyKit:/bin/false
postfix:x:51:51:Postfix Daemon:/var/spool/postfix:/bin/false
sshd:x:71:65:SSH daemon:/var/lib/sshd:/bin/false
wwwrun:x:30:8:WWW daemon apache:/var/lib/wwwrun:/bin/false
vscan:x:65:110:Vscan account:/var/spool/amavis:/bin/false
```

## b. /etc/nologin

Antes de pasar por un seudoshell de conexión, los módulos PAM permiten aplicar numerosas limitaciones. Entre ellas, el módulo `pam_nologin`, que le permite prohibir la conexión de los usuarios excepto root. Este módulo monitoriza si un usuario intenta conectarse al fichero `/etc/nologin`. Es útil para tareas de mantenimiento, ya que sólo root se puede conectar.

```
$ pwd
/etc/pam.d
$ grep nologin *
login:auth      requisite    pam_nologin.so
ppp:auth       required     pam_nologin.so
sshd:auth      requisite    pam_nologin.so
```

Piense también que puede prohibir el acceso de una cuenta dada vía el módulo PAM `pam_listfile`.

## c. /etc/security

En el mismo estilo, el fichero `/etc/securetty` contiene la lista de los terminales considerados como seguros. Para el servicio dado, se prohibirá la conexión si el terminal de la persona que intenta conectarse no está incluido. En el ejemplo siguiente, se autorizan los logins (vía el comando `login`) únicamente desde los pseudoterminales locales, es decir, sólo desde las consolas directamente accesibles en el ordenador vía las combinaciones [Alt][F1] a [Alt][F7].

```
$ grep securetty *
login:auth      [user_unknown=ignore success=ok ignore=ignore
auth_err=die default=bad]      pam_securetty.so
$ cat /etc/securetty
#
# This file contains the device names of tty lines (one per line,
# without leading /dev/) on which root is allowed to login.
#
tty1
tty2
tty3
tty4
tty5
tty6
```



¡No confunda `securetty` (Secure tty) con `security`!

## 6. Probar las contraseñas

Las herramientas crack y «John the ripper» intentan encriptar sus contraseñas, tanto desde un diccionario como por la fuerza bruta (unas tras otras). En el peor de los casos, las encuentran en unos segundos; en el mejor, en varios días, incluso semanas. Si esto es así, se puede considerar que la contraseña es buena.

«John the ripper» es muy fácil de usar. El comando es **john**. Para probar la integridad de su fichero, `/etc/shadow`:

```
# john /etc/shadow
Loaded 5 password hashes with 5 different salts (OpenBSD Blowfish [32/64])
```

En el modo por defecto, `john`:

- intenta una detección simple vía combinaciones corrientes relacionadas con la cuenta,
- pasa al modo diccionario con aplicación de reglas,
- luego, intenta una búsqueda incremental.

¡Una búsqueda puede tardar de unos segundos a varias semanas!

Para probar un solo usuario:

```
# john -user:seb /etc/shadow
```

Para probar los usuarios con un diccionario:

```
# john -users:seb -wordlist:/var/lib/john/wordlists/all /etc/shadow
```

Lo mismo pero probando varias reglas por palabra (inversión, mayúsculas, minúsculas, etc.):

```
# john -users:seb -wordlist:/var/lib/john/wordlists/all -rules /etc/shadow
```

Para retomar una búsqueda interrumpida ([Ctrl][C]):

```
# john -restore
```

John ubica sus resultados en el directorio `~/.john`. Generalmente sólo root debe utilizar esta herramienta:

```
# ls -l .john
total 4
-rw----- 1 root root 70 may 23 21:58 john.pot
-rw----- 1 root root 124 may 23 21:54 john.rec
```

El

fichero `john.pot` contiene los resultados encontrados por John. El fichero no está vacío y esto indica que hay un problema: john ha encontrado una contraseña. El fichero `john.rec` contiene el estado actual de la búsqueda por si se produce una interrupción y es utilizado por el programa para reanudar la búsqueda.

En una máquina basada en Intel Core 2 duo de 64 bits a 3,4 GHz, se cראה la contraseña de Seb (el autor) en 4 minutos y 22 segundos. Estaba (de manera voluntaria esta vez) basada en una palabra del diccionario.

Si usted pulsa una tecla durante la búsqueda, John muestra su estado.

Suponiendo una cuenta Enrique cuyo propietario tiene como contraseña la misma palabra que su nombre de cuenta, a saber «Enrique»:

```
# john -users:enrique /etc/shadow
Created directory: /root/.john
Loaded 1 password hash (OpenBSD Blowfish [32/64])
enrique          (enrique)
guesses: 1  time: 0:00:00:00 100% (1)  c/s: 4.34  trying: enrique
```

Eso debería hacer reflexionar a los que piensan que nadie vendrá a molestarle y debería animarlos a modificar su contraseña con reglas precisas (mayúsculas, minúsculas, cifras, etc.).

## 7. Buscar rootkits

### a. Fundamentos del rootkit

Una vez que un pirata informático haya conseguido, vía un fallo o una contraseña demasiado sencilla, penetrar en su máquina, intentará seguramente instalarse una «puerta mayor de entrada», o más bien una puerta trasera, una **backdoor**, con el fin de poder volver a utilizar su máquina con propósitos dudosos, o llevárselo o almacenar datos (de esta manera, algunos PC se han encontrado con miles de ficheros mp3 o divx, o han sido utilizados como servidor por personas poco escrupulosas).

Lo ideal para el pirata es poder acaparar los derechos de root. Es su objetivo: ser el dueño total de su máquina. Para ello, no siempre le hace falta «teclar» directamente en esta cuenta. En Linux, es habitual (e incluso aconsejable) conectarse como simple usuario y luego de cambiar a root el tiempo de efectuar las operaciones necesarias. Ahora bien, para pasar a root, puede utilizar **su**.

Si, vía el servicio ftp mal configurado o vía ssh (y luego scp), el pirata intenta conectarse a una cuenta cuya contraseña es obvia (como en el ejemplo de la cuenta enrique), y consigue desplegar en el sistema un script llamado «su» y modifica el PATH por defecto para poner en él la ruta al script, entonces, es pan comido.

```
$ pwd
/home/seb
$ cat su
#!/bin/bash
echo -e "Contraseña :\c"
read -s password
echo "$@ $password" > /tmp/fic
```

```
echo
echo "su: Contraseña incorrecta."
/bin/su $@
$ chmod +x su
$ export PATH=$HOME:$PATH
$ su - root
Contraseña: ===> FALSO SU
su: Contraseña incorrecta. ===> FALSO SU
Contraseña: ===> VERDADERO SU
```

Sólo falta mostrar el contenido del fichero para obtener la contraseña:

```
$ cat /tmp/fic
- root azerty
```

El

método, simplista pero eficaz, no es imparable ni mucho menos: su sólo pide la contraseña una vez, y a no ser que se sea muy despistado, el engaño salta a la vista. Al hecho de ubicar scripts, modificar el entorno, sustituir un fichero por otro de modo que se obtenga un acceso privilegiado a una máquina, se le denomina instalar un **rootkit**. Una vez en su sitio, este último garantiza, mientras no se detecte, un acceso root a la máquina.

## b. chkrootkit

La herramienta **chkrootkit** es una herramienta sencilla que permite buscar la presencia de los rootkits más conocidos y habituales. Sólo es eficaz si se actualiza y se ejecuta regularmente; no sustituye a los controles ya nombrados anteriormente.

```
# chkrootkit
ROOTDIR is ''
Checking 'amd'... not found
Checking 'basename'... not infected
Checking 'biff'... not found
Checking 'chfn'... not infected
Checking 'chsh'... not infected
Checking 'cron'... not infected
Checking 'crontab'... not infected
...
Searching for sniffer's logs, it may take a while... nothing found
Searching for HiDrootkit's default dir... nothing found
Searching for t0rn's default files and dirs... nothing found
Searching for t0rn's v8 defaults... nothing found
Searching for Lion Worm default files and dirs... nothing found
Searching for RSHA's default files and dir... nothing found
Searching for RH-Sharpe's default files... nothing found
Searching for Ambient's rootkit (ark) default files and dirs... nothing found
Searching for suspicious files and dirs, it may take a while...
...
Searching for LPD Worm files and dirs... nothing found
Searching for Ramen Worm files and dirs... nothing found
Searching for Maniac files and dirs... nothing found
Searching for RK17 files and dirs... nothing found
Searching for Ducoci rootkit... nothing found
Searching for Adore Worm... nothing found
Searching for ShitC Worm... nothing found
Searching for Omega Worm... nothing found
Searching for Sadmind/IIS Worm... nothing found
...
Searching for ENYELKM rootkit default files... nothing found
Searching for anomalies in shell history files... Warning: '' is
linked to another file
```

```
Checking 'asp'... not infected
Checking 'bindshell'... not infected
Checking 'lkm'... chkproc: nothing detected
Checking 'rexedcs'... not found
Checking 'sniffer'... eth0: not promisc and no PF_PACKET sockets
vmnet8: not promisc and no PF_PACKET sockets
vmnet1: not promisc and no PF_PACKET sockets
Checking 'w55808'... not infected
...
```

## 8. Los virus

Ya han aparecido los primeros virus en Unix. Aunque se suele dotar de mayor seguridad al sistema y aunque los virus en las plataformas Unix y Linux (incluyendo Mac OS X) son casi inexistentes, en el caso más o menos probable de que un virus penetre y comprometa la seguridad de su máquina, las de otros o la de toda la red, usted tiene la responsabilidad de erradicarlo.

Si su máquina es un servidor, en particular de correo electrónico o de ficheros en una red que contiene máquinas con Windows muy expuestas, usted no debe servir de vector indirecto de propagación. Tiene que eliminar la amenaza.

Existe varios antivirus en Linux, algunos comerciales (gratuitos o no), otros libres. El antivirus Clam (ClamAV) es libre y gratuito. Está disponible en la dirección <http://www.clamav.net/>. Se actualiza cada día.

**Freshclam** permite actualizar las bases de firmas, ubicadas en `/var/lib/clamav`:

```
# pwd
/var/lib/clamav
# freshclam
ClamAV update process started at Mon Dec 26 10:31:22 2011
Downloading main.cvd [100%]
main.cvd updated (version: 54, sigs: 1044387, f-level: 60, builder: sven)
Downloading daily.cvd [100%]
daily.cvd updated (version: 14193, sigs: 54465, f-level: 63, builder: guitar)
Downloading bytecode.cvd [100%]
bytecode.cvd updated (version: 158, sigs: 38, f-level: 63, builder: edwin)
Database updated (1098890 signatures) from database.clamav.net (IP: 217.173.238.34)
# ls -l
total 31360
-rw-r--r-- 1 clamav clamav 56880 dic 26 10:32 bytecode.cvd
-rw-r--r-- 1 clamav clamav 1298399 dic 26 10:32 daily.cvd
-rw-r--r-- 1 clamav clamav 30750647 dic 26 10:32 main.cvd
-rw----- 1 clamav clamav 52 dic 26 10:32 mirrors.dat
```

**Clamscan** permite buscar los posibles virus:

```
# clamscan -v /usr/local/bin
Scanning /usr/local/bin/clview
/usr/local/bin/clview: OK
Scanning /usr/local/bin/fclauer
/usr/local/bin/fclauer: OK
Scanning /usr/local/bin/unmakeclos
/usr/local/bin/unmakeclos: OK
Scanning /usr/local/bin/makeclos
/usr/local/bin/makeclos: OK
Scanning /usr/local/bin/clwblock
/usr/local/bin/clwblock: OK
Scanning /usr/local/bin/firefox-uninstall-pkcs11.sh
/usr/local/bin/firefox-uninstall-pkcs11.sh: OK
Scanning /usr/local/bin/clpasswd
/usr/local/bin/clpasswd: OK
Scanning /usr/local/bin/clexport
```

```
/usr/local/bin/clexport: OK
Scanning /usr/local/bin/cldel
/usr/local/bin/cldel: OK
Scanning /usr/local/bin/firefox-install-pkcs11.sh
/usr/local/bin/firefox-install-pkcs11.sh: OK
Scanning /usr/local/bin/clls
/usr/local/bin/clls: OK
Scanning /usr/local/bin/clmakefs
/usr/local/bin/clmakefs: OK
Scanning /usr/local/bin/climport
/usr/local/bin/climport: OK
```

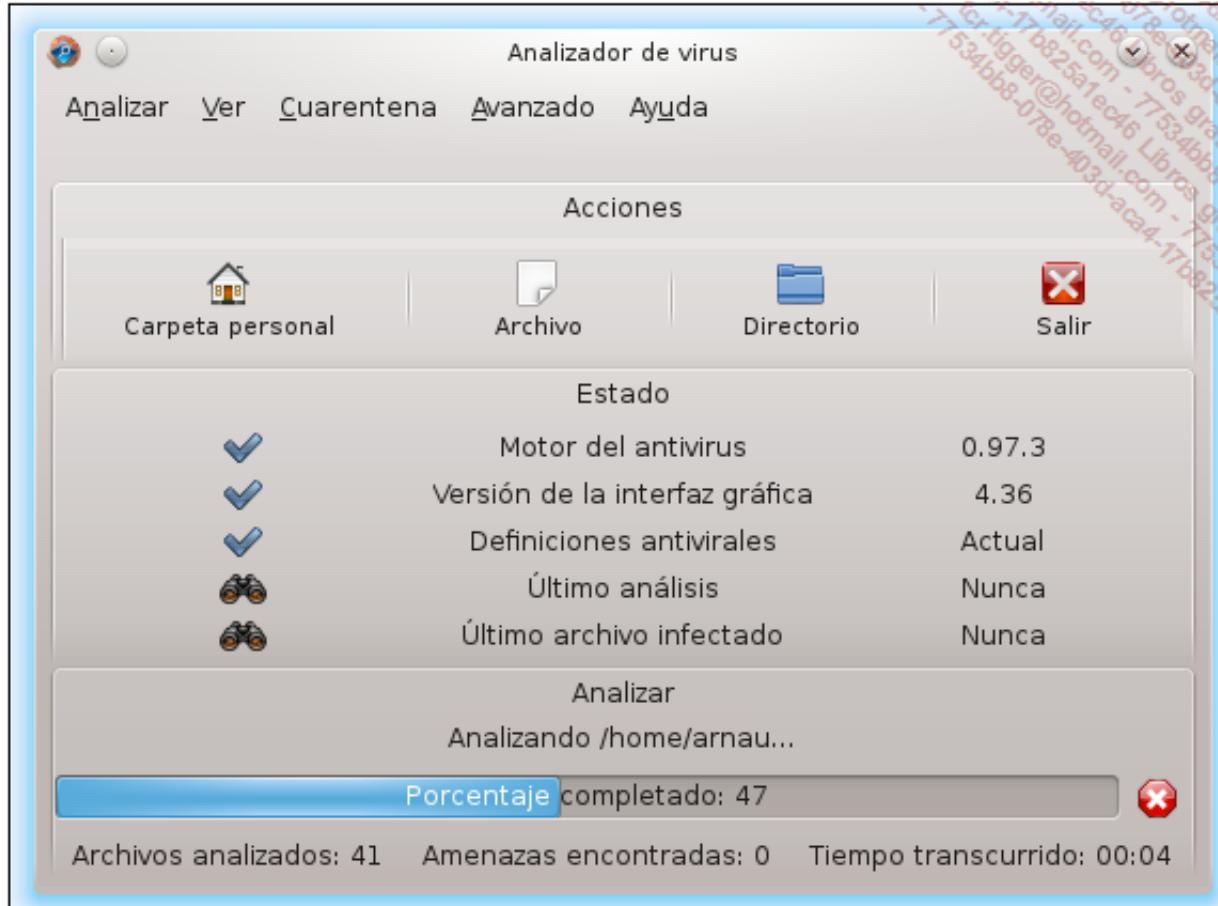
```
----- SCAN SUMMARY -----
Known viruses: 1097623
Engine version: 0.97.3
Scanned directories: 1
Scanned files: 13
Infected files: 0
Data scanned: 0.49 MB
Data read: 0.48 MB (ratio 1.01:1)
Time: 2.063 sec (0 m 2 s)
```

continuación, la misma prueba con un virus falso bajo tres formas: binario, comprimido gzip y comprimido bzip2. Si se detecta un virus, se mueve el fichero correspondiente a `/home/seb/VIRUS`:

```
$ clamscan -v -r --move=/home/seb/VIRUS /home/seb/bin
Scanning /home/seb/bin/eicarcom2.zip
/home/seb/bin/eicarcom2.zip: Eicar-Test-Signature FOUND
/home/seb/bin/eicarcom2.zip: moved to '/home/seb/VIRUS//eicarcom2.zip'
Scanning /home/seb/bin/eicar_com.zip
/home/seb/bin/eicar_com.zip: Eicar-Test-Signature FOUND
/home/seb/bin/eicar_com.zip: moved to '/home/seb/VIRUS//eicar_com.zip'
Scanning /home/seb/bin/eicar.com
/home/seb/bin/eicar.com: Eicar-Test-Signature FOUND
/home/seb/bin/eicar.com: moved to '/home/seb/VIRUS//eicar.com'
```

```
----- SCAN SUMMARY -----
Known viruses: 1097623
Engine version: 0.97.3
Scanned directories: 1
Scanned files: 13
Infected files: 0
Data scanned: 0.49 MB
Data read: 0.48 MB (ratio 1.01:1)
Time: 2.063 sec (0 m 2 s)
```

Se puede iniciar Clamav como servicio. En este caso, se puede configurar la búsqueda de virus en estructuras concretas, lo que permite la existencia de herramientas tales como clamtk o la integración de funciones en gestores de archivos (como Nautilus en Gnome).



*Analizando la carpeta personal con ClamTk.*

## 9. Los límites del usuario

El campo de acción de los PAM es más amplio que la simple conexión, ya que gestiona también el entorno del usuario. Incluso antes de ver el módulo en cuestión, el comando **ulimit** permite actuar en el entorno del shell y de los procesos que controla. El parámetro **-a** muestra las opciones controladas por **ulimit**:

```
$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority     (-e) 0
file size               (blocks, -f) unlimited
pending signals          (-i) 16380
max locked memory       (kbytes, -l) 32
max memory size        (kbytes, -m) 1753125
open files              (-n) 1024
pipe size               (512 bytes, -p) 8
POSIX message queues    (bytes, -q) 819200
real-time priority       (-r) 0
stack size               (kbytes, -s) 8192
cpu time                (seconds, -t) unlimited
max user processes      (-u) 16380
virtual memory          (kbytes, -v) 3333600
file locks               (-x) unlimited
```

Las líneas en negrita merecen su atención.

- **max memory size**: el tamaño de memoria máximo que puede ocupar el usuario;
- **open files**: el número máximo de descriptores de ficheros; por lo tanto, el número máximo de ficheros que se pueden abrir;
- **max user processes**: el número máximo de procesos que puede iniciar un usuario.

Se pueden cambiar estos valores según ciertos límites impuestos por el administrador. Existen límites soft (suaves o bajos) que son los valores por defecto devueltos por ulimit, y límites hard (duros, altos), que no se pueden superar.

Para pasar el número máximo de ficheros abiertos a 2048:

```
$ ulimit -n 2048
$ ulimit -n
2048
```

El

administrador root puede controlar los valores por defecto gracias al fichero `/etc/security/limits.conf`.

```
$ grep seb /etc/security/limits.conf
seb          hard    nproc        32768
seb          soft    nofile      1024
seb          hard    nofile      4096
```

En este ejemplo, el usuario Seb está limitado a un máximo de 32.768 procesos, puede abrir por defecto 1.024 ficheros, pero puede montar 4.096 mediante una acción ulimit por su parte.

## 10. Los derechos SUDO

### a. Dar privilegios extendidos

El comando **sudo** permite asignar el derecho a ejecutar comandos de administrador a uno o varios usuarios, en una o varias máquinas. En la práctica, para que un usuario pueda ejecutar un comando que, en principio, sólo puede ejecutar root, debe añadirse un derecho sudo a este usuario para este comando.

El fichero de configuración de sudo es `/etc/sudoers`. Es posible editarlo a mano o con el comando **visudo**. Este último comando comprueba la sintaxis del fichero en el momento de guardar.

La clásica sintaxis de una línea sudo es la siguiente:

```
user      machine = (user2) comando
```

- **user**: el usuario (o alias) al cual se aplica la regla.
- **máquina**: la máquina (o el alias) en la cual se aplica la regla.
- **user2**: la cuenta con la cual el usuario ejecutará el comando.
- **comando**: el comando que se va a ejecutar.

Por ejemplo, la línea siguiente va a autorizar al usuario Seb a ejecutar el comando `fsck` y sus parámetros con los derechos root sobre cualquier máquina (donde esté presente esta regla):

```
seb      ALL = /sbin/fsck
```

Para utilizar **fsck**, Seb debe usar el comando **sudo** como a continuación:

```
seb@slyserver:~/handbrake> sudo /sbin/fsck  
seb's password:  
fsck 1.41.1 (01-Sep-2008)  
e2fsck 1.41.1 (01-Sep-2008)  
...
```

Por defecto, se pide la contraseña del usuario Seb antes de proseguir. El usuario puede obtener la lista de sus derechos sudo:

```
seb@slyserver:~/handbrake> sudo -l  
User seb may run the following commands on this host:  
(root) /sbin/fsck
```

Por defecto, se autentica el usuario una primera vez; luego, no se vuelve a pedir su contraseña mientras continúe en la misma sesión (mientras no cierre su consola o su entorno).

La ventaja de **sudo**, además de asignar derechos puntuales a un grupo de personas determinadas, es la trazabilidad. Se transmiten los mensajes de **sudo** a **syslog**, que los puede volver a dirigir a un fichero. Puede ser `/var/log/messages`:

```
May  8 14:39:08 slyserver sudo:      seb  
: TTY=pts/3 ; PWD=/home/seb ; USER=root ; COMMAND=/sbin/fsck  
May  8 14:40:14 slyserver sudo:      seb  
: TTY=pts/3 ; PWD=/home/seb ; USER=root ; COMMAND=list
```

Se puede establecer el registro de destino mediante `syslog.conf` o `syslog_ng.conf`.

## b. Sintaxis de `/etc/sudoers`

El ejemplo anterior es deliberadamente limitado. Es posible:

- crear grupos de usuarios,
- crear grupos de máquinas,
- crear grupos de comandos,
- forzar o no el uso de una contraseña,
- forzar la ejecución de un comando bajo un usuario distinto de root.

Los grupos se llaman alias.

Para crear alias de usuarios, utilice esta sintaxis:

```
User_Alias      ADMINS = seb, esteban, enrique
```

Si todos los administradores deben poder utilizar el comando **fsck**, la línea se convierte en:

```
ADMINS      ALL= /sbin/fsck
```

Si los administradores sólo pueden ejecutar estos comandos en determinadas máquinas, cree un alias de máquinas:

```
Host_Alias      SERVERS= slyserver, eepc
```

La línea sudo se convierte en:

```
ADMINS      SERVERS= /sbin/fsck
```

Es posible añadir varios comandos unos tras otros, con o sin parámetros. Si los ADMINS deben ejecutar también el comando `/sbin/dumpe2fs`, la línea se convierte en lo siguiente:

```
ADMINS SERVERS=/sbin/fsck, /sbin/dumpe2fs
```

o, para saltar a la línea siguiente, ponga una barra oblicua «\» en el final de línea:

```
ADMINS SERVERS=/sbin/fsck, \
/sbin/dumpe2fs
```

Puede crear alias de comandos para agruparlos:

```
Cmnd_Alias ADMCMD=/sbin/fsck, /sbin/dumpe2fs
```

La línea sudo se convierte en:

```
ADMINS SERVERS=ADMCMD
```

Para evitar la introducción de la contraseña de un usuario, escriba NOPASSWD como a continuación:

```
ADMINS SERVERS=NOPASSWD: ADMCMD
```

Los usuarios del alias ADMINS ya no tendrán que teclear contraseña para introducir los comandos del alias ADMCMD.

También puede forzar la utilización de una contraseña con PASSWD. Añada así el comando mkfs:

```
ADMINS SERVERS=NOPASSWD: ADMCMD, PASSWD:/sbin/mkfs
```

Para permitir que los ADMINS inicien un comando con la identidad de otro usuario distinto de root, sitúe el nombre del usuario (o alias) entre paréntesis, como a continuación:

```
ADMINS ALL=(esteban) PASSWD: /sbin/service
```

ADMINS podrá ejecutar `/sbin/service` como Esteban introduciendo la contraseña de éste.

El resultado del `sudo -l` asociado es el siguiente:

```
seb@slyserver:~> sudo -l
User seb may run the following commands on this host:
  (root) NOPASSWD: /sbin/fsck, /sbin/dumpe2fs
  (root) /sbin/mkfs
  (esteban) /sbin/service
```

Para iniciar el comando bajo otro usuario, utilice el parámetro `-u` de sudo:

```
seb@slyserver:~> sudo -u esteban /sbin/service
```

Puede precisar varios usuarios, pero también crear alias como para el resto:

```
Runas_Alias LISTUSR=seb,esteban
```

Puede utilizar este alias en la línea sudo:

```
ADMINS ALL=(LISTUSR) PASSWD: /sbin/service
```

Puede utilizar el alias ALL: define el conjunto de usuarios, máquinas y comandos, según la posición donde está escrito. La línea siguiente significa que los ADMINS tienen permiso de ejecutar todos los comandos, en todas las máquinas, sin contraseña:

```
ADMINS ALL=NOPASSWD: ALL
```

El signo de exclamación permite excluir a determinados usuarios, comandos, máquinas, de un alias o de una lista. La línea siguiente da todos los derechos sobre todos los programas sin contraseña a los ADMINS, salvo sobre /sbin/mkfs, que requiere una contraseña, y /sbin/resize2fs, que está prohibido:

```
ADMINS SERVERS=NOPASSWD: ALL, !/sbin/resize2fs, PASSWD:/sbin/mkfs
```

Si un ADMINS intenta iniciar `resize2fs`, obtiene un error:

```
seb@slyserver:~> sudo /sbin/resize2fs  
Sorry, user seb is not allowed to execute '/sbin/resize2fs' as root on slyserver.
```

La regla siguiente, muy peligrosa, autoriza a todo el mundo a ejecutar lo que quiera, en cualquier sitio y sin contraseña, como cualquier usuario:

```
ALL      ALL=(ALL) NOPASSWD: ALL
```

A continuación viene una línea que se utiliza con frecuencia para autorizar a un usuario a hacerlo todo, salvo iniciar shells o el comando **su**, con la condición de que los alias SU y SHELLS estén completos:

```
seb      ALL = ALL, !SU, !SHELLS
```

Esta línea no garantiza que Seb pueda obtener derechos por medios indirectos, por ejemplo, si se han vuelto a nombrar todos los comandos, o por rebotes de comandos sucesivos.

## 11. Auditoría más completa

Para efectuar la auditoría de un sistema, puede utilizar, además del acceso a los rastros e históricos, y de los comandos ya listados, los productos libres o gratuitos como tripwire, que comprueba la integridad del sistema; COPS, que vigila la seguridad del sistema, y Crack, que detecta las contraseñas erróneas.

## 12. Los boletines de seguridad

### a. CERT: Computer Emergency Response Team

#### Historia

Se debe a un estudiante de la universidad de Cornell el primer virus capaz de replicarse sobre la red Internet (se llamaba Arpanet en esa época). Desarrollado y lanzado a finales de 1998 sin intención de perjudicar, este programa se propagaba y replicaba sólo aprovechando fallos de seguridad de Unix y sus servicios. Este programa saturó rápidamente la red Internet y las máquinas que había alcanzado, paralizando Internet, compuesta entonces de 60 000 ordenadores, durante varios días. Sólo se había afectado al 4% de las máquinas.

Para erradicar este virus se necesitó tiempo y muchos medios suministrados por el MIT, Berkeley, etc. Hubo que estudiar su funcionamiento para entender cómo se había comportado y sólo entonces se pudieron corregir los agujeros de seguridad de los sistemas y servidores. Se aplicaron parches correctivos. El DARPA, iniciador del proyecto Arpanet (luego Internet) introdujo más tarde una nueva estructura llamada el CERT (CERT/CC, *CERT Coordination Center*), encargada de analizar las amenazas futuras y la vulnerabilidad de los sistemas.

Internet representa decenas de millones de máquinas interconectadas, con sistemas operativos y diferentes

servicios. El CERT (<http://www.cert.org/>), cuya sede se encuentra a día de hoy en la universidad de Carnegie Mellon, sigue estudiando las posibles vulnerabilidades en Internet, y ello incluso a largo plazo, con el fin de obtener la mejor seguridad posible.

## **Papel del CERT**

Las misiones del CERT son las siguientes:

- centralización y análisis de las peticiones de asistencia después de los incidentes de seguridad (ataques) sobre las redes y los sistemas de información: recepción de las peticiones, análisis de los síntomas y correlación de los incidentes;
- tratamiento de las alertas y reacción a los ataques informáticos: análisis técnico, intercambio de información con otros CERT, contribución a estudios técnicos específicos;
- establecimiento y mantenimiento de una base de datos de las vulnerabilidades;
- prevención por publicación de información sobre las precauciones que es preciso tomar para minimizar los riesgos de incidente, o a lo peor sus consecuencias;
- posible coordinación con otras entidades: centros de competencia de redes, operadores y proveedores de acceso a Internet, CERT nacionales e internacionales.

## **Boletines del CERT**

Existen varios tipos de comités CERT: por país, por industria, etc. Se emiten boletines independientes, pero vinculados entre ellos. A continuación indicamos dos sitios donde encontrar alertas:

<https://www.ccn-cert.cni.es>

Y sobre todo, <http://www.us-cert.gov/>

The screenshot shows a Mozilla Firefox browser window with the following details:

- Title Bar:** US-CERT Technical Cyber Security Alert TA08-137A - OpenSSL Random Number Generator Vulnerability - Mozilla Firefox
- Address Bar:** http://www.us-cert.gov/castechalerts/TA08-137A.html
- Header:** Thursday, September 03, 2009 | Home | FAQ | Contact | Privacy & Use
- Logo:** US-CERT logo featuring the Department of Homeland Security seal.
- Section:** National Cyber Alert System
- Section:** Technical Cyber Security Alert TA08-137A
- Section:** Debian/Ubuntu OpenSSL Random Number Generator Vulnerability
- Text:** Original release date: May 16, 2008  
Last revised: --  
Source: US-CERT
- Section:** Systems Affected
  - Debian, Ubuntu, and Debian-based distributions
- Section:** Overview
- Text:** A vulnerability in the OpenSSL package included with the Debian GNU/Linux operating system and its derivatives may cause weak cryptographic keys to be generated. Any package that uses the affected version of SSL could be vulnerable.
- Section:** I. Description
- Text:** A vulnerability exists in the random number generator used by the OpenSSL package included with the Debian GNU/Linux, Ubuntu, and other Debian-based operating systems.

Un ejemplo de alerta es un fallo de seguridad de la librería OpenSSL en Debian *Debian/Ubuntu OpenSSL Random Number Generator Vulnerability*, Referencia TA08-137A. Las claves aleatorias generadas desde la versión OpenSSL de Debian y Ubuntu no lo son realmente, lo que limita enormemente la seguridad de estas llaves y conlleva un problema, ya que la actualización de la librería no es suficiente; también hay que generar de nuevo las claves existentes...

## b. SecurityFocus

**SecurityFocus** (anteriormente Bugtraq) es una lista de difusión electrónica (e-mailing) en línea desde 1993 que agrupa discusiones sobre vulnerabilidades, seguridad, anuncios y medios para detectar fallos y cómo corregirlos. Originalmente la lista nació de la necesidad de erradicar dos problemas:

- los numerosos fracasos del CERT en prevenir problemas;
- la desidia de numerosos editores y constructores, que no suministraban las actualizaciones de seguridad a pesar de los fallos encontrados.

Bugtraq pertenecía a SecurityFocus, que a su vez pertenece hoy al editor de las suites de seguridad de Symantec.

Puede darse de alta en la mailing-list desde el sitio <http://www.securityfocus.com/archive>.

The screenshot shows the homepage of SecurityFocus. At the top, there's a banner for "Symantec Connect" with the subtext "A technical community for Symantec customers, end-users, developers, and partners." Below the banner, there's a "Join the conversation >" button. The main content area is titled "Vulnerabilities" and lists several security issues:

Vulnerability Title	Date	Description
Apache Tomcat HTML Manager Interface HTML Injection Vulnerability	2011-12-23	OpenStack Nova Image Registration Arbitrary Input Validation Vulnerability
Apache Tomcat HTTP DIGEST Authentication CVE-2011-1184 Multiple Security Weaknesses	2011-12-23	Mozilla Firefox/Thunderbird/SeaMonkey SVG Animation Elements Information Disclosure Vulnerability
Apache Tomcat SecurityManager Security Bypass Vulnerability	2011-12-23	Mozilla Firefox/Thunderbird/SeaMonkey YARR Library Denial Of Service Vulnerability
Apache Tomcat 'MemoryUserDatabase' Information Disclosure Vulnerability	2011-12-23	Mozilla Firefox and Thunderbird CVE-2011-3666 Remote Code Execution Vulnerability
International Components for Unicode '_canonicalize()' Memory Corruption Vulnerability	2011-12-23	Mozilla Firefox/SeaMonk CVE-2011-3664 NULL Pointer Dereference Denial Of Service Vulnerability

At the bottom of the list, there's a link "» Search all vulnerabilities".

Página de inicio de SecurityFocus

## c. Los boletines de las distribuciones

Los editores de las mayores distribuciones facilitan también boletines de seguridad. Reutilizan a su vez las

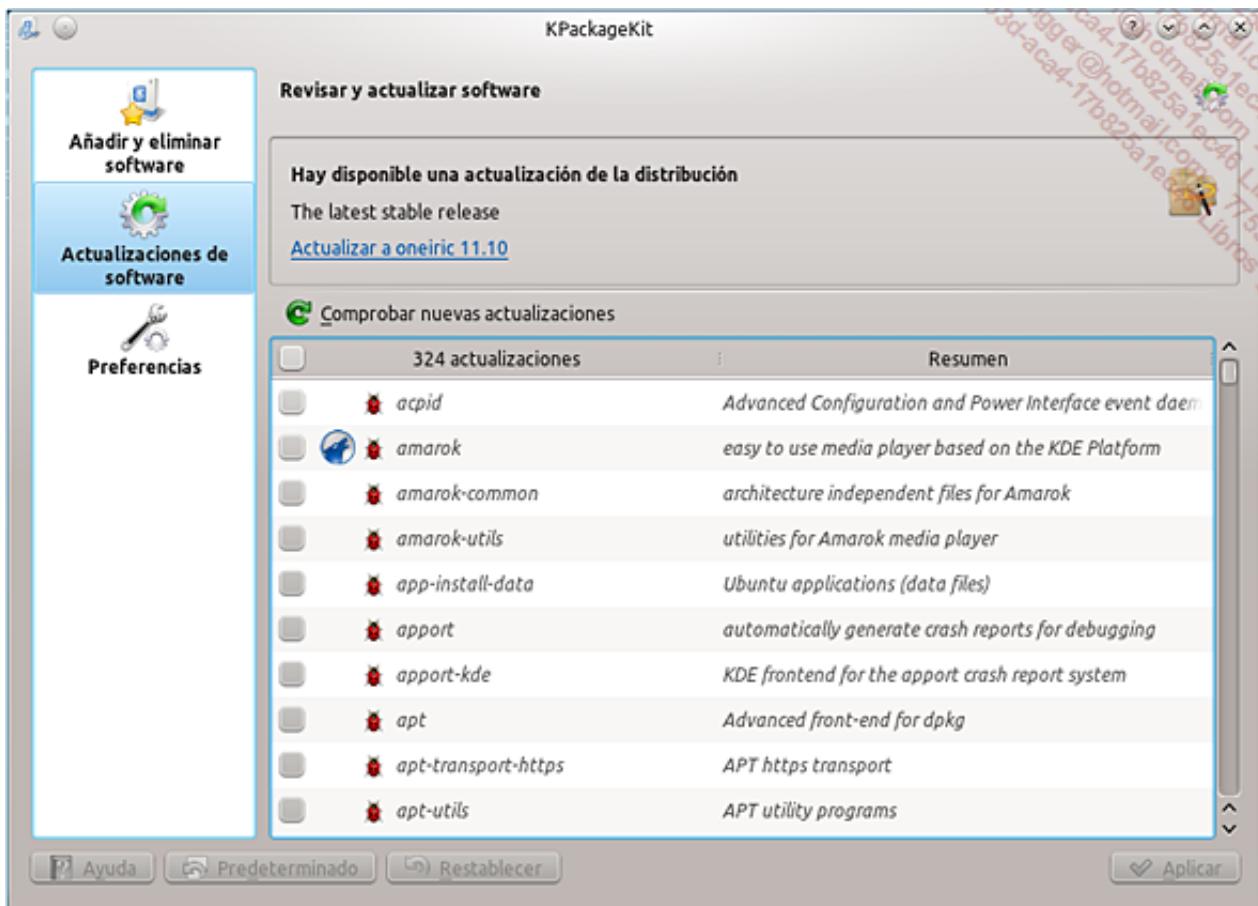
alertas de seguridad de otros organismos como el CERT o las listas de Bugtraq, pero, dado que cada editor es responsable de los paquetes que publica, debe publicar él mismo la correspondiente corrección, y es habitual que parchee algunos productos y éstos difieran del original. De ahí la emisión de una alerta para prevenir a sus clientes y usuarios. A continuación mostramos los sitios donde puede obtener información de seguridad para las principales distribuciones:

- Debian: <http://www.debian.org/security/>
- openSUSE: <http://lists.opensuse.org/opensuse-security-announce/>
- Fedora: <https://www.redhat.com/archives/fedora-security-list/>
- Ubuntu: <http://www.ubuntu.com/usn>
- Mandriva: <http://www.mandriva.com/security/advisories>
- Red Hat Enterprise: <http://www.redhat.com/security/updates/errata/>

#### **d. Los parches correctores**

No basta con dar una alerta cuando se detecta un agujero de seguridad; también hace falta arreglarlo. Para ello, los editores suministran o bien paquetes corregidos (el vínculo se encuentra a menudo en el boletín de alerta), o bien parches correctores. Las distribuciones suministran habitualmente un componente que permite recuperar estos parches e informarle sobre su disponibilidad.

Cada distribución dispone de una herramienta que permite comprobar e informar al usuario de las actualizaciones del sistema. Esta herramienta puede ser distinta según la distribución; puede consultar la documentación proporcionada por el autor de la distribución para conocerla. Sin embargo, parece que el uso de PackageKit (<http://www.packagekit.org/>) es cada vez el más común, por lo menos su módulo de actualización. Éste es el caso de Ubuntu, openSUSE (y los productos Novell) y Fedora. Iniciado en su interfaz gráfica, le informa por la barra de tareas o la zona de notificación de las actualizaciones disponibles para su entorno. El siguiente ejemplo muestra la ejecución de kPackageKit (módulo de PackageKit para KDE) anunciando la presencia de un parche de seguridad, en Ubuntu 11.04.



*Detección de una actualización de seguridad por PackageKit*

# Seguridad de servicios y de red

## 1. Comprobar los puertos abiertos

### a. Los sockets

Las conexiones de red entre dos máquinas se efectúa por **sockets**. Un socket es una conexión entre dos puntos mediante una red. Una máquina dispone de una dirección IP y de puertos (virtuales) de conexión numerados, a los cuales se vinculan servicios (ver el capítulo La red). Un cliente establece una conexión desde un puerto de su máquina (número de puerto > 1024, en general elegido de manera aleatoria entre los puertos libres) hacia un puerto determinado de otra máquina, por ejemplo un servidor Web sobre el puerto 80. La comunicación establecida entre los dos pasa por un socket. Se puede configurar el sistema para aceptar o rechazar conexiones desde o hacia determinados puertos locales o distantes; ídem para las direcciones IP. Es el papel del firewall (cortafuegos) como, por ejemplo, **Netfilter**.

En una instalación Linux típica no se activan los cortafuegos, excepto si dispone de esa opción durante la instalación. No se filtran los puertos y cada máquina exterior puede intentar establecer una conexión a la suya a través de un puerto: a esto se denomina «abrir un socket».

Eso no significa forzadamente que haya un agujero de seguridad: si no hay servicio alguno a la escucha, la conexión es imposible. En situación normal los equipos inician muchos servicios por defecto. Si algunos padecen vulnerabilidades, o se han configurado de manera demasiado laxa, se produce un riesgo real de intrusión.

### b. Información desde netstat

El capítulo La red le ha presentado la herramienta **netstat**, que permite obtener información y estadísticas de red sobre una máquina local. En particular se puede comprobar cuáles son los puertos a la escucha en su máquina, quién ha establecido una conexión y qué procesos locales (servicios) están a la escucha:

```
# netstat -a -A inet -p
Conexiones Internet activas (servidores y establecidas)
Proto Recv-Q Send-Q Dirección local           Dirección remota
Estado      PID/Program name
tcp        0      0 *:5800                  *:*
LISTEN     31232/kded [kdeinit
tcp        0      0 *:5801                  *:*
LISTEN     23224/xinetd
tcp        0      0 *:vnc-server            *:*
LISTEN     31232/kded [kdeinit
tcp        0      0 *:5901                  *:*
LISTEN     23224/xinetd
tcp        0      0 *:sunrpc                *:*
LISTEN     3076/portmap               *:*
tcp        0      0 *:ndmp                  *:*
LISTEN     26746/sshd
tcp        0      0 *:6000                  *:*
LISTEN     31088/Xorg
tcp        0      0 *:7634                  *:*
LISTEN     3006/hddtemp
tcp        0      0 *:ftp                   *:*
LISTEN     26772/vsftpd
tcp        0      0 localhost:ipp            *:*
LISTEN     3092/cupsd
tcp        0      0 localhost:smtp            *:*
LISTEN     3159/master
tcp        0      0 slyserver:38144          eeepr:ssh
ESTABLISHED 25622/ssh
tcp        0      0 slyserver:45046          eeepr:ms-wbt-server
ESTABLISHED 7803/rdesktop
tcp        0      0 slyserver:ftp            eeepr:49502
FIN_WAIT2   -
udp        0      0 *:xdmcp                *:*
    27850/kdm
udp        0      0 *:59742                *:*
    2984/avahi-daemon:
udp        0      0 *:mdns                 *:*
    2984/avahi-daemon:
udp        0      0 *:sunrpc                *:*
```

Las líneas en negrita muestran tres conexiones establecidas o terminadas entre las máquinas slyserver y eeepr.

- el eeepr estaba conectado en el puerto ftp de slyserver,
- slyserver está conectado en el puerto ssh del eeepr,
- slyserver está conectado al puerto ms-wbt-server del eeepr (protocolo terminal server).

```

3076/portmap
udp      0      0 *:ipp
*:*      3092/cupsd

```

### c. La herramienta nmap

Existe un batallón de herramientas de seguridad, de comprobación, de pruebas de fiabilidad, etc. La herramienta **nmap** forma parte de él. Se define como una herramienta de exploración de red y de control de seguridad. Permite probar las conexiones de red de una máquina determinada y obtener mucha información. En particular, analizando tramas, logra determinar el tipo y la versión del sistema operativo remoto.

El manual de instrucciones de nmap tiene más de 2000 líneas, por lo que es imposible estudiarlo todo. A continuación mostramos algunas de sus posibilidades:

Examine los puertos a la escucha en la máquina de prueba que se ha usado durante la redacción de este libro. Varios puertos están abiertos (unos servicios están a la escucha). Algunos de ellos pueden presentar riesgos: telnet, netbios-ssn y microsoft-ds (recursos compartidos de Windows), ftp, vnc, etc.

```

# nmap localhost
Starting Nmap 4.20 ( http://insecure.org ) at 2008-05-22 14:41 CEST
Interesting ports on localhost (127.0.0.1):
Not shown: 1684 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
25/tcp    open  smtp
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
631/tcp   open  ipp
5800/tcp  open  vnc-http
5801/tcp  open  vnc-http-1
5900/tcp  open  vnc
5901/tcp  open  vnc-1
6000/tcp  open  X11
10000/tcp open  snet-sensor-mgmt

Nmap finished: 1 IP address (1 host up) scanned in 0.170 seconds

```

En algunos casos, es posible desactivar los servicios no necesarios:

En cuanto se detiene un servicio de red, el puerto ya no está accesible.

El parámetro `-A` permite detectar además el sistema operativo remoto y su versión. Para esto, uno o varios puertos deben estar abiertos. Más aún; nmap interroga a cada servicio asociado a los puertos encontrados para recuperar información. Abajo y en negrita resaltamos los valores más relevantes:

Esto quiere decir que una máquina dispone de muy pocos secretos si no está bien protegida. El servidor encontrado tiene como sistema operativo Linux Debian Etch, que dispone de un núcleo comprendido entre 2.6.14 y 2.6.17 (un error, porque el núcleo realmente es un 2.6.18). Los servicios con los que cuenta son:

- OpenSSH 4.3p2
- Apache 2.2.3 y PHP

```

# chkconfig telnet off
# service vsftpd stop
# service smb stop
# service nmb stop
# service xinetd restart
# nmap localhost
Starting Nmap 4.20 ( http://insecure.org ) at 2008-05-22 14:48 CEST
Interesting ports on localhost (127.0.0.1):
Not shown: 1688 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
111/tcp   open  rpcbind
631/tcp   open  ipp
5800/tcp  open  vnc-http
5801/tcp  open  vnc-http-1
5900/tcp  open  vnc
5901/tcp  open  vnc-1
6000/tcp  open  X11
10000/tcp open  snet-sensor-mgmt

```

```

# nmap -A machine
Starting Nmap 4.20 ( http://insecure.org ) at 2008-05-22 19:54 CEST
Interesting ports on machine.midominio.com (192.168.1.25):
Not shown: 1676 closed ports
PORT      STATE SERVICE      VERSION
9/tcp     open  discard
13/tcp    open  daytime
21/tcp    open  ftp          vsftpd 2.0.5
22/tcp    open  ssh          OpenSSH 4.3p2 Debian 9 (protocol 2.0)
25/tcp    open  smtp         Postfix smptd

```

```

37/tcp  open  time      (32 bits)
53/tcp  open  domain
80/tcp  open  http      Apache httpd 2.2.3 ((Debian) PHP/5.2.0-8+etch10)
111/tcp open  rpcbind   2 (rpc #100000)
113/tcp open  ident     OpenBSD identd
139/tcp open  netbios-ssn Samba smbd 3.X (workgroup: SLYNET)
199/tcp open  smux      Linux SNMP multiplexer
445/tcp open  netbios-ssn Samba smbd 3.X (workgroup: SLYNET)
606/tcp open  mountd    1-2 (rpc #100005)
631/tcp open  ipp       CUPS 1.2
901/tcp open  http      Samba SWAT administration server
1389/tcp open  ldap      OpenLDAP 2.2.X
2049/tcp open  nfs      2 (rpc #100003)
3128/tcp open  squid-http
7937/tcp open  nsreexec 1 (rpc #390113)
7938/tcp open  rpcbind   2 (rpc #100000)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.14 - 2.6.17
Uptime: 112.678 days (since Wed Jan 30 21:39:53 2008)
Network Distance: 2 hops
Service Info: Host: machine.midominio.org; OSs: Unix, Linux, OpenBSD

OS and Service detection performed. Please report any incorrect results
at http://insecure.org/nmap/submit/ .
Nmap finished: 1 IP address (1 host up) scanned in 115.360 seconds

```

Basta con consultar los boletines de seguridad para ver si una de estas versiones es conocida por tener problemas de seguridad; si es el caso, su servidor presenta riesgos.

## 2. Suprimir los servicios inútiles

### a. Cuestiones generales

Si alguien «hackea» su sistema, es que hay alguien malintencionado que ha encontrado la manera de entrar. En contra de la idea extendida, la instalación de un firewall no resuelve todos los problemas: más aún cuando, en un puesto de trabajo, la tendencia es abrir varios puertos de redes hacia Internet (o mejor dicho, desde Internet): ftp, http, p2p (redes eDonkey, o sea eMule, BitTorrent, etc.), ssh y así sucesivamente. Ahora bien, con que un único servicio iniciado tenga un agujero de seguridad es suficiente para que su máquina sea atacada y aumenten sus problemas.

Incluso aunque el servicio no sea muy conocido, es la posible que la configuración que usted haya aplicado sea demasiado sencilla o laxa. No sería muy inteligente dejar que su servidor ssh aceptara las conexiones desde el exterior si su contraseña o la del root es joselito, password o algo similar. Durante un ataque, el autor de este libro tuvo la oportunidad de constatar que el atacante intentaba conectarse en bucle, vía ssh, utilizando una serie de logins/contraseñas predefinidos entre varias combinaciones clásicas preconfiguradas por defecto en determinados programas. Un caso sencillo es el de una instalación de MySQL por defecto donde la cuenta de administración no tiene contraseña.

Evalúe desactivar todos los servicios que no necesita. Si resulta que necesita algunos en ciertos momentos, y no en otros, no dude en iniciarlos sólo cuando los necesite para pararlos después. Asimismo, en su firewall (netfilter o otro), deje abiertos sólo los puertos estrictamente necesarios.

### b. Servicios standalone

Los servicios standalone, o sea iniciados de manera independiente, están controlados mediante el comando **service** o **/etc/init.d/service**. Para controlar las paradas y reinicio de estos servicios de manera permanente, utilice los comandos **chkconfig** (distribuciones RPM) o **rcupdate.d** (Debian).

### c. Servicios xinetd

Se pueden activar y desactivar los servicios controlados por xinetd con la opción **disable** de su fichero de configuración.

```

$ pwd
/etc/xinetd.d
$ grep disable *
chargen:      disable      = yes
chargen-udp:  disable      = yes
cups-lpd:     disable      = yes
...
vmware-authd: disable      = no
vnc:          disable      = yes
...

```

Para actualizar los cambios, fuerce **xinetd** para que vuelva a cargar su configuración.

```
# /etc/init.d/xinetd
```

### 3. Los tcp\_wrappers

Las **envolturas TCP** o simplemente **tcp\_wrappers** permiten la comprobación de los accesos a un servicio de red determinado (service, xinetd, portmapper). Cada programa que utiliza los **tcp\_wrappers** se compila con la librería **libwrap** de manera estática (el comando ldd no permite ver la librería).

Para saber si un servicio de red está compilado con libwrap, se introduce el comando siguiente:

```
strings -f <binario> | grep hosts_access
```

Aquí tenemos un ejemplo con **xinetd**, que utiliza las **tcp\_wrappers**:

```
# strings -f /usr/sbin/xinetd |grep hosts_access  
/usr/sbin/xinetd: hosts_access
```

Si no se devuelve ninguna línea, el programa no utiliza las **tcp\_wrappers**.

Entre los servicios que utilizan las **tcp\_wrappers**, encontramos:

- **sendmail** (incluyendo postfix),
- **sshd** (ssh),
- **xinetd** (y por lo tanto de manera indirecta todos los servicios asociados),
- **vsftpd** (ftp),
- **portmap** (y por lo tanto nis, nfs),
- **in.telnetd** (telnet), así como la mayoría de los servicios soportados por xinetd,
- **dovecot** (imap, pop).

La comprobación de acceso a un servicio envuelto TCP se hace en tres etapas:

- ¿se autoriza el acceso de manera explícita?
- si no es el caso, ¿se prohíbe el acceso de manera explícita?
- si no es el caso, por defecto, se autoriza el acceso.

Los ficheros de configuración son **/etc/hosts.allow** y **/etc/hosts.deny**. La sintaxis es común:

```
daemon_list: client_list [:options]
```

- **daemon\_list**: lista de los **ejecutables (NO DE LOS SERVICIOS)** separados por comas. Puede poner **ALL** para especificar todos los servicios. Si se dispone de varias interfaces red, se puede usar la sintaxis con @: servicio@ip.

```
in.telnetd: ...  
sshd, portmap: ...  
sshd@192.168.1.7: ...
```

- **client\_list**: clientes autorizados o prohibidos para este servicio. Se puede especificar la dirección IP, el nombre, la máscara de red, el nombre de la red, etc.

```
... : 192.168.1.7, 192.168.1.8  
... : 192.168.1.  
... : puesto1, puesto2  
... : 192.168.1.0/255.255.255.0  
... : .midominio.org
```

La lista de clientes admite una sintaxis avanzada:

- **ALL**: correspondencia sistemática.
- **LOCAL**: todos los anfitriones cuyo nombre no contiene punto (puesto1, puesto2, etc.).
- **UNKNOWN**: anfitrión cuyo nombre no se puede resolver.
- **KNOWN**: anfitrión cuyo nombre se puede resolver.
- **PARANOID**: anfitrión cuyo nombre no se puede resolver o cuyo IP no tiene resolución inversa.

- **EXCEPT:** permite excluir ciertos anfitriones.

ALL EXCEPT puesto10

Para verificar una regla, el sistema lee primero `/etc/hosts.allow`, luego `/etc/hosts.deny`. La búsqueda se detiene a la primera correspondencia encontrada. Una línea en hosts.allow autoriza la conexión. Una línea en hosts.deny prohíbe la conexión. Si no se deniega de manera explícita el acceso, se autoriza: la petición no corresponde a ningún criterio.

En el ejemplo siguiente:

- Sólo los miembros de la subred 192.168.1.0 tienen permiso para conectarse al servidor ftp (prohibido para todos los demás).
- Los anfitriones puesto 1 y puesto 2 tienen acceso a telnet y portmap.
- Los anfitriones de baddominio.org, excepto trusted, no tienen conexión alguna posible.
- Se prohíbe el servicio pop/imap a todos los de la red 192.168.0.0, salvo 192.168.1.5.

```
# /etc/hosts.allow
vsftpd:          192.168.1.
in.telnetd, portmap:  puesto1, puesto2

# /etc/hosts.deny
ALL: .baddominio.org except trusted.baddominio.org
vsftpd,in.telnetd,portmap: ALL
dovecot : 192.168.0. EXCEPT 192.168.1.5
```

## 4. Netfilter

### a. Presentación

Netfilter es una arquitectura de filtro de los paquetes para los núcleos de Linux 2.4 y 2.6. El filtrado se hace en el mismo núcleo al nivel de las capas 2, 3 y 4 del modelo OSI, es decir, los vínculos dados, red y transporte. Por ejemplo, es capaz de actuar a bajo nivel en las interfaces ethernet (2), en la pila IP (3) y en los protocolos de transporte como TCP (4). El filtrado no tiene estado: como Netfilter sólo inspecciona los encabezamientos de los paquetes, es muy veloz y no conlleva tiempo de espera.

Se pueden inspeccionar los contenidos de los paquetes (protocolos aplicativos) usando extensiones, pero este trabajo se delega a herramientas de usuario.

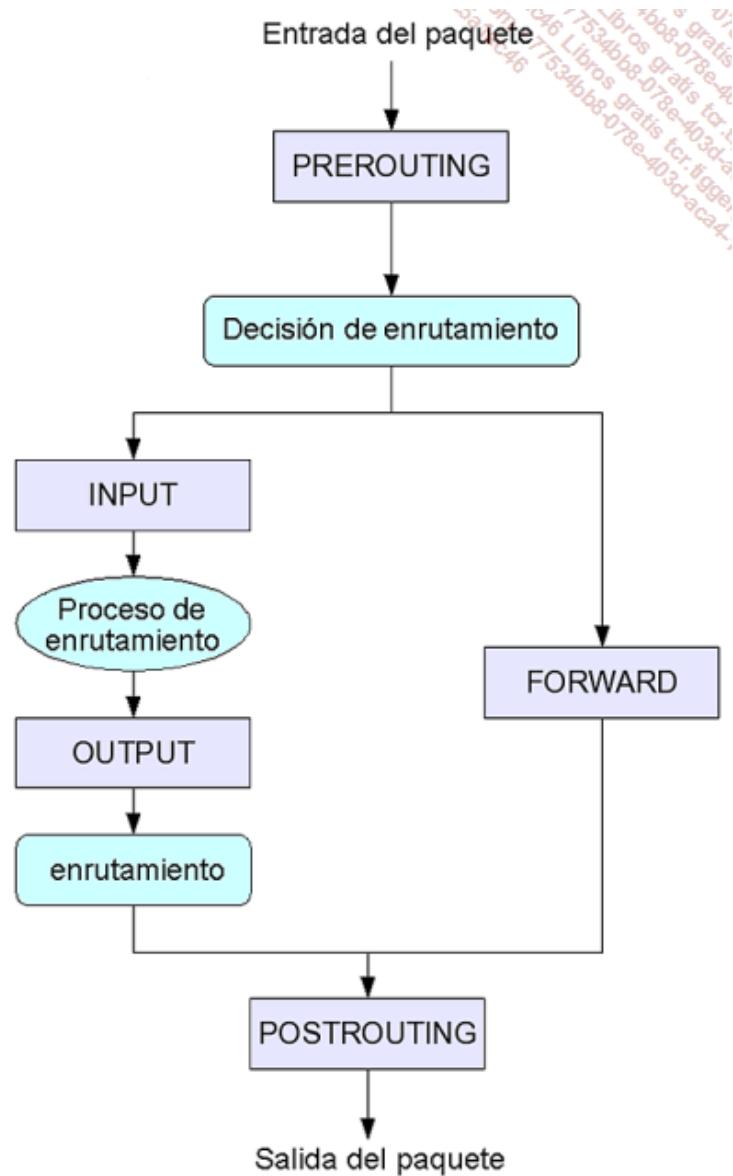
Dicho de otro modo, netfilter es un firewall que actúa al nivel del protocolo.

El programa usuario que permite actuar sobre las reglas de filtrado es **iptables**.

Se realiza la implementación al nivel del núcleo con módulos.

### b. Vida de un paquete

Mejor un esquema que un largo discurso. El paquete llega por arriba y sale por abajo. Entre ambos puntos, pasa por diferentes niveles de netfilter.



*Ciclo de vida de un paquete de red con netfilter.*

Cada estado (rectángulo) corresponde a un punto de filtrado posible mediante el comando **iptables**.

- **PREROUTING:** trata los paquetes a su llegada. Si un paquete tiene como destino el sistema local, se le tratará con un proceso local (INPUT, OUTPUT). En caso contrario, y si está activado el forwarding, se aplicarán las reglas FORWARD y POST\_ROUTING.
- **FORWARD:** los paquetes sólo cruzan el sistema local. Trata los paquetes enrutados a través del sistema local.
- **INPUT:** trata los paquetes destinados al sistema local, en entrada (después del enrutamiento).
- **OUTPUT:** trata los paquetes que dejan el sistema local, antes POSTROUTING.
- **POSTROUTING:** trata los paquetes justo antes de su salida del sistema.

### c. Principio de las reglas

Cuando netfilter trata un paquete, lo relaciona con un determinado número de reglas que determinan lo que se debe hacer con él.

- Se ordenan las reglas: la posición de una regla en una lista indica si se utilizará la regla y cuándo se hará.
- Se prueban los paquetes con cada una de las reglas, una tras otra.

- Netfilter funciona según el mecanismo de la primera correspondencia. Si una regla corresponde, se desatienden las otras y se detiene la comprobación.
- Una regla puede especificar varios criterios.
- Para que una regla corresponda a un paquete, todos los criterios deben corresponder.
- Si a pesar de todas las reglas el paquete pasa, se puede aplicar una regla por defecto.

#### d. Destinos de reglas

Un destino de regla determina qué acción se debe emprender cuando un paquete corresponde a los criterios de una regla. Se utiliza la opción **-j** de **iptables** para especificar el destino.

Los dos destinos básicos son DROP y ACCEPT. netfilter cuenta con extensiones que añaden otros destinos, como LOG o REJECT.

- **DROP**: se rechaza el paquete. No se manda ninguna notificación a la fuente.
- **REJECT**: se rechaza el paquete. Notifica un error a la fuente.
- **ACCEPT**: se acepta el paquete.
- **LOG**: se manda una información de seguimiento al syslog.

Puede crear reglas sin destino. En este caso, la regla incrementará el recuento de un contador de paquetes y un contador de bytes asociados a la regla con objetivos estadísticos.

#### e. Primer ejemplo

A continuación se presenta una regla sencilla que prohíbe los paquetes procedentes de 192.168.1.11.

```
# iptables -A INPUT -s 192.168.1.11 -j DROP
```

- **-A**: punto de filtrado (INPUT, OUTPUT, FORWARD, PREROUTING, POSTROUTING), llamado también **cadena**.
- **-s**: fuente, puede ser una dirección IP, un nombre de anfitrión, una red, etc.
- **-j**: jump, objetivo o destino de la regla (ACCEPT, DROP, REJECT...)

Resultado: usted prohibirá la entrada de los paquetes cuya fuente es 192.168.1.11.

#### f. Operaciones básicas

Se numeran las reglas a partir de 1.

- Añada una regla con **-A**.

```
# iptables -A INPUT -s 192.168.1.2 -j DROP
```

- Inserte una regla con **-I** en la posición deseada.

```
# iptables -I OUTPUT -d 192.168.1.25 -j DROP 3 # insertar a la 3a posición
```

- Suprima una regla con **-D**.

```
# iptables -D INPUT 1 # suprime la regla 1
```

- Liste las reglas con **-L**.

```
# iptables -L OUTPUT
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
Chain FORWARD (policy ACCEPT)
target     prot opt source          destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
```

- Utilice **-F** (flush) para suprimir el conjunto de las reglas.  

```
# iptables -F
```
- Utilice **-P** (policy) para modificar las reglas por defecto de una cadena.

```
# iptables -P INPUT DROP
# iptables -L INPUT
Chain INPUT (policy DROP)
target     prot opt source          destination
```

## **g. Criterios de correspondencia**

### **General**

Los criterios de correspondencia determinan la validez de una regla. Se deben comprobar todos los criterios para bloquear un paquete. Los criterios básicos son:

- **-i**: interfaz entrante (filtrado de capa 2);
  - **-o**: interfaz saliente (filtrado de capa 2);
  - **-p**: protocolo de capa 4. Para los nombres, ver el fichero `/etc/protocols`;
  - **-s**: dirección IP de la fuente (o red);
  - **-d**: dirección IP de destino (o red).
- Prohibir las entradas por eth0.  

```
iptables -A INPUT -i eth0 -j DROP
```
  - Prohibir el forward entre eth1 y eth2.  

```
iptables -A FORWARD -i eth1 -o eth0 -j DROP
```
  - Prohibir el protocolo ICMP en entrada (iel ping!).  

```
iptables -A input -p icmp -j DROP
```

### **TCP, UDP e ICMP**

Siguiendo el protocolo (capa 4), algunas opciones son posibles. Es el caso de tcp, udp o icmp (en particular utilizado por ping). Se realiza en general el filtrado a nivel de protocolo con extensiones a netfilter.

- **-p**: protocolo (tcp, udp, icmp, etc.)
- **--sport**: puerto fuente
- **--dport**: puerto destino

Si desea prohibir, por ejemplo, las conexiones entrantes con destino al puerto 80 (servidor httpd), proceda como a continuación:

```
iptables -A INPUT -p tcp --dport 80 -j DROP
```

### **Argumentos de los criterios**

Para las direcciones, puede especificar:

- un anfitrión por su nombre o su dirección IP;
- una red por su nombre o su máscara (192.168.1.0/24, 192.168.1.0/255.255.255.0).

Para los puertos:

- un número;

- un nombre (ver/etc/servicios);
- una gama de puertos: **123:1024**.

En todos estos casos puede usarse el signo de exclamación para definir excepciones.

Para prohibir en la entrada todas las conexiones salvo las de 10.0.0.1:

```
# iptables -A INPUT -s ! 10.0.0.1 -j DROP
```

### **Usuarios y grupos**

Netfilter gestiona los UID y GID de los procesos que emiten paquetes desde el host. Puede realizar un filtrado en la salida para los usuarios y grupos locales, para las reglas OUTPUT y POSTROUTING. El módulo es **owner**. Los usuarios se especifican con **--uid-owner** y los grupos con **-gid-owner**.

Esto puede ser muy útil, por ejemplo, en el caso de un proxy transparente. Se puede redirigir cualquier conexión saliente con destino el puerto 80 hacia el puerto 3128 para todo usuario excepto el proxy. Para ello hay que ejecutar el siguiente comando:

```
# iptables -t nat -A OUTPUT -m owner ! --uid-owner proxy -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 3128
```

Para permitir a un usuario pasar por otra regla, puede ejecutar el siguiente comando ANTES (recuerde que el orden es importante) de la redirección:

```
# iptables -t nat -A OUTPUT -p tcp --dport 80 -m owner --uid-owner seb -j ACCEPT
```

### **h. Tablas**

Además de los puntos de filtrado (INPUT, OUTPUT, etc.) Netfilter define tablas particulares que pueden contener reglas específicas. Puede crear sus propias tablas que contengan sus propias reglas y a continuación adjuntarlas a un punto de filtrado. Pero Netfilter también dispone de tablas por defecto y especialmente la tabla **nat**, que contiene las reglas que permiten modificar las IP y puertos origen y destino de los paquetes. Se muestra del siguiente modo:

```
# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source          destination
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
ACCEPT    tcp  --  anywhere        anywhere
tcp dpt:www owner UID match root
ACCEPT    tcp  --  anywhere        anywhere
tcp dpt:www owner UID match seb
ACCEPT    tcp  --  anywhere        anywhere
tcp dpt:www owner UID match esteban
REDIRECT  tcp  --  anywhere        anywhere
! owner UID match proxy tcp dpt:www redir ports 3128
...
```

La siguiente regla modifica dinámicamente la IP de destino de un cierto tipo de paquetes. Los paquetes con destino el puerto 80 de la IP 192.168.1.2, se modificarán de tal modo que se reenviarán a la 192.168.10.10:  
# iptables -t nat -j

### **i. Guardar las configuraciones**

Las reglas definidas con **iptables** se cargan en memoria y se transmiten de manera dinámica al núcleo. Pero al arrancar de nuevo la máquina, se pierden. **Red Hat** permite guardar el conjunto de las reglas de manera que se vuelvan persistentes.

```
# service iptables save
```

Las reglas están guardadas en el fichero **/etc/sysconfig/iptables**.

Para las otras distribuciones, consulte la documentación. En algunos casos, puede ser útil crear su propio script.

Se deberían cargar las reglas **iptables** ANTES de la activación de la red. Así, no hay ningún riesgo a nivel de seguridad, porque las reglas serán directamente válidas en el momento en que se active la red.

## **5. UFW**

Partiendo de la premisa que iptables es un poco complicado para el usuario medio o incluso un poco avanzado, los desarrolladores de la distribución Ubuntu han decidido crear una herramienta en línea de comandos que permita configurar de forma más sencilla Netfilter. UFW (*Uncomplicated Firewall*) es una capa frontal de Netfilter e iptables. No los reemplaza, sino que permite crear en su lugar reglas Netfilter. Utiliza y es un complemento de iptables. Sólo hay que usar un comando: **ufw**.

UFW está disponible para Ubuntu y también puede instalarse en Debian. Si no está presente por defecto en otras distribuciones, se puede recomilar, las fuentes están disponibles. Se están dedicando esfuerzos para crear paquetes, especialmente para Fedora. Los comandos siguientes se deben ejecutar directamente como root o desde sudo.

## a. Activación y estado

Active ufw con el parámetro **enable**, como se muestra a continuación:

```
# ufw enable  
El cortafuegos está activo y habilitado en el arranque del sistema
```

En Ubuntu, ufw se inicia mediante init. El script de init comprobará el

estado de activación de ufw para determinar si debe cargar las reglas o no hacerlo. De este modo, el script init siempre debe estar activo, pero es esta llamada al comando ufw la que determina si debe funcionar o no.

El estado actual se obtiene con el parámetro **status**. En el ejemplo siguiente ya hay una regla presente (la de acceso a través del puerto ssh):

```
# ufw status  
Estado: activo  
  
Hasta Acción Desde  
---- ----  
22 ALLOW Anywhere
```

Se puede añadir el

parámetro **verbose** para obtener más detalles, especialmente para mostrar el estado por defecto de las distintas tablas:

```
# ufw status verbose  
Estado: activo  
Acceso: on (low)  
Por defecto: deny (Entrada), allow (Salida)
```

ufw se desactiva con disable. Las reglas creadas con ufw se descargará. Sin embargo, las reglas creadas por iptables no se modificarán.

```
# ufw disable  
El cortafuegos está detenido y desactivado en el arranque del sistema
```

## b. Reglas por defecto

Con ufw, siendo voluntariamente sencillo, sólo el tráfico entrante y saliente tiene reglas por defecto: allow o deny para los permisos, como en los ejemplos siguientes:

```
# ufw default deny incoming  
La política incoming predeterminada cambió a «deny»  
(asegúrese de actualizar sus reglas consecuentemente)  
# ufw default allow outgoing  
La política outgoing predeterminada cambió a «allow»  
(asegúrese de actualizar sus reglas consecuentemente)
```

El registro se activa y se desactiva del siguiente modo:

```
# ufw logging off  
Registro desactivado  
# ufw logging on  
Registro activado
```

## c. Gestión de reglas

### Reglas simples

La forma más sencilla consiste en permitir o prohibir conexiones entrantes o salientes en función de los puertos. Por ejemplo, puede permitir el acceso al servicio ssh de su servidor:

```
# ufw allow 22  
Regla añadida
```

se define en /etc/services.

Para prohibir el acceso con deny:

```
# ufw deny 22  
Regla actualizada
```

que este comando no elimina la regla antigua, la actualiza. Para eliminar definitivamente una regla, muéstrelas todas de forma numérica, como se muestra a continuación:

```
# ufw status numbered  
Estado: activo  
  
 Hasta          Acción      Desde  
 ----          -----  
 [1] Anywhere    ALLOW IN   212.27.38.253/udp  
 [2] 22           DENY IN    Anywhere
```

```
# ufw delete 2  
Borrando:  
deny 22  
¿Continuar con la operación (s|n)? s  
Regla eliminada
```

```
# ufw delete deny 22  
Regla eliminada
```

un servidor Samba requiere la apertura de algunos puertos. En la instalación del paquete asociado, los scripts de postinstalación llamarán a ufw para definir un perfil que usted podrá activar a continuación. La lista de aplicaciones para las que hay reglas se muestra con **app list**:

```
# ufw app list  
Aplicaciones disponibles:  
CUPS  
OpenSSH  
Postfix  
Postfix Submission  
Samba  
Squid
```

```
# ufw app info Samba  
Perfil: Samba  
Título: LanManager-like file and printer server for Unix  
Descripción: The Samba software suite is a collection of programs that implements the SMB/CIFS protocol for unix systems, allowing you to serve files and printers to Windows, NT, OS/2 and DOS clients. This protocol is sometimes also referred to as the LanManager or NetBIOS protocol.
```

```
Puertos:  
137,138/udp  
139,445/tcp
```

```
# ufw allow Samba  
Regla añadida  
# ufw deny Samba  
Regla actualizada
```

## **Reglas más complejas**

Para cada puerto, puede especificar el protocolo:

Mejor que el número del puerto, puede utilizar el nombre del servicio tal y como

### **Eliminación**

Hay veces en que no nos es suficiente, ya

Elimine la regla 2:

También puede borrar la regla explícitamente:

### **Aplicaciones**

Debido a que ufw tiene que ser sencillo a ojos del usuario, las aplicaciones instaladas pueden ofrecer reglas predefinidas para simplificar la integración de reglas asociadas. Por ejemplo, el uso de

Los detalles de una aplicación se muestran con app info. Por ejemplo, consultamos los detalles de Samba:

Las reglas se activan o se desactivan fácilmente, tal y como se muestra a continuación:

```
# ufw allow 22/tcp
```

También puede utilizar una sintaxis más extendida, como la que prohíbe todo el tráfico entrante en TCP desde la subred 192.168.1.0/24 hacia cualquier IP de su servidor el puerto 22:

```
# ufw deny proto tcp from 192.168.1.0/24 to any port 22
```

La regla siguiente permitirá el acceso a los puertos 80, 443 y del 8080 al 8090 en su ordenador:

```
# ufw allow proto tcp from any to any port 80,443,8080:8090
```

Para prohibir cualquier conexión desde su equipo hacia cualquier equipo al puerto 80:

```
# ufw reject out to any port 80
```

## 6. GPG

### a. Un clon de PGP

**GPG** (*Gnu Privacy Guard*) es un clon libre de **PGP** (*Pretty Good Privacy*). Implementa el algoritmo de cifrado RSA. PGP es objeto de una norma que GPG respeta. Esto significa que las dos implementaciones son compatibles: las llaves generadas por el uno o el otro se pueden intercambiar.

El objetivo de GPG es cifrar una comunicación gracias a un cifrado por claves asimétricas. Una clave permite firmar el texto, otra clave sirve para encriptar el texto.

El cifrado por claves asimétricas utiliza dos claves, una pública y otra privada:

- Su **clave pública**, difundida públicamente, permite al destinatario descifrar el mensaje que usted ha firmado (con la clave privada).
- Su **clave privada** le permite firmar un mensaje. La persona que lo recibe comprueba su firma con la ayuda de la clave pública que usted le ha proporcionado, demostrando así que usted es realmente el autor del mensaje.

A continuación vamos a presentar todo lo necesario para crear y gestionar un almacén de llaves (también llamado llavero): sus claves públicas y privadas, las claves públicas de sus amigos, las firmas, etc.

Se presentan los comandos en modo consola. Existen herramientas gráficas, como **kgpg**, para gestionar las claves. Asimismo, numerosos clientes de mensajería integran GPG (Kmail lo integra por defecto). Gracias al plugin Enigmail se puede incorporar fácilmente GPG a Thunderbird. Se utilizan también las firmas para comprobar el origen de los paquetes RPM, por ejemplo.

### b. Generar las claves

Utilice `gpg` con el parámetro `--gen-key`. GPG empieza con crear, si no existen ya, el directorio raíz de GPG propio de cada usuario y los ficheros que contendrán los elementos gestionados por gpg.

```
seb@slyserver:~> gpg --gen-key
gpg (GnuPG) 2.0.9; Copyright (C) 2008 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: directorio '/home/seb/.gnupg' creado
gpg: nuevo fichero de configuración '/home/seb/.gnupg/gpg.conf' creado
gpg: ADVERTENCIA: las opciones de '/home/seb/.gnupg/gpg.conf' aún no están
activas esta vez
gpg: el llavero '/home/seb/.gnupg/secring.gpg' ha sido creado
gpg: el llavero '/home/seb/.gnupg/pubring.gpg' ha sido creado
```

Seleccione el tipo de clave deseado. La primera elección incluye las otras dos; por lo tanto, es la que debería efectuarse:

Luego debe elegir el tamaño, en bits, de la clave que hay que generar. Cuanto más larga es la clave, más complejo es el cifrado. Pero en máquinas más antiguas, el desciframiento puede tardar

```
Seleccione el tipo de clave deseado:
(1) DSA y Elgamal (por defecto)
(2) DSA (firma sola)
(5) RSA (firma sola)
¿Su elección? 1
```

entonces más tiempo. Puede elegir una clave de 1024 bits por ejemplo:

El par de claves DSA dará 1024 bits.  
las claves ELG pueden tener entre 1024 y 4096 bits de longitud.  
¿Qué tamaño de clave desea? (2048) 1024  
El tamaño pedido es 1024 bits

Luego indique la duración de validez de las claves. Por defecto, si prevee utilizarla el mayor tiempo posible, elija

cero para una duración infinita. Si no, precise la duración deseada según el formato indicado. Valide su elección. Debe saber que es posible invalidar una clave generada por error con una clave de revocación de la que hablaremos un poco más adelante.

Especifique cuánto tiempo debería ser válida esta clave.

0 = la clave no expira  
<n> = la clave expira en n días  
<n>w = la clave expira en n semanas  
<n>m = la clave expira en n meses  
<n>y = la clave expira en n años

¿La clave es válida para? (0)

La clave no expira.

¿Es correcto? (s/N) s

Para generar la clave, GPG le pide introducir su nombre, su dirección de correo electrónico (e-mail) y un comentario. En el nombre, no dude en poner sus apellidos y su nombre. Se podrá

utilizar luego el comentario como alias para determinados comandos GPG. No será posible cambiar más adelante estos elementos. Por el contrario, le será posible añadir direcciones de email con el fin de asociar varias de ellas a una sola clave.

Necesita un nombre de usuario para identificar su clave, el programa lo construye a partir del nombre real, de un comentario y de una dirección de email, de la manera siguiente:  
«Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>»

Nombre real: Sébastien ROHAUT

Dirección de e-mail: sebastien.rohaut@dominio.es

Comentario: slyce

Usted utiliza el juego de caracteres 'utf-8'.

Ha seleccionado este nombre de usuario:

"Sébastien ROHAUT (slyce) <sebastien.rohaut@dominio.es>"

Cambiar el (N)ombre, el (C)omentario, el (E)-mail o (O)K/(S)alir ? 0

La **passphrase** o frase de contraseña es complementaria a la clave privada. Se pedirá esta frase a cada envío o recepción de mensaje privado encriptado. También asegura, al menos mientras no la encuentren, una cierta seguridad de la clave privada, incluso

si la roban, ya que esta clave no se puede utilizar sin esta frase. Elija una frase (o palabra) bastante compleja, que recuerde siempre, pero no muy larga, ya que tendrá que teclearla de manera regular.

Necesita una frase de contraseña para proteger su clave secreta.

Inserte la frase de contraseña:

Repita la frase de contraseña:

GPG efectúa la última etapa, la generación de las propias claves. GPG va a utilizar

información introducida y genera números aleatorios. Le damos un consejo: durante el cálculo, haga trabajar su ordenador con el fin de «sacudir» el generador de números aleatorios.

Se debe generar un gran número de bytes aleatorios. Tendría que hacer otra cosa (teclar, mover el ratón, utilizar discos) durante la generación de los primeros números. Eso da al generador de nombres aleatorios más oportunidad de tener bastante entropía.

gpg: clave 13E021A8 marcada como si tuviera una confianza última.  
se ha creado y firmado las claves pública y secreta.

gpg: comprobar la base de confianza

gpg: 3 marginale(s) necesarias, 1 completa(s) necesarias, modelo de confianza PGP

gpg: profundidad: 0 válida: 1 firmado: 0

confianza: 0-. 0g. 0n. 0m. 0f. 1u

pub 1024D/13E021A8 2009-05-04

Huella de la clave = 6115 DE46 2678 40AB 0AB2 B8B9 1F12 D427 13E0 21A8

uid Sébastien ROHAUT (Slyce) <sebastien.rohaut@dominio.es>

sub 1024g/BA311C5C 2009-05-04

Las últimas cuatro líneas resumen toda la información relativa al trabajo efectuado por GPG:

- **pub:** da la longitud de la clave (1024 bits), de número 13E021A8, generada el 04/05/2009.

- **huella:** es el «fingerprint» que permite determinar la validez de la clave pública, un poco

como la suma md5 de cualquier fichero. Es casi imposible (pero no del todo), que claves de gran tamaño tengan una huella idéntica.

- **uid**: información nominativa (nombre, comentario o alias/seudo, dirección e-mail).
- **sub**: es el tamaño (1024 bits), el número y la fecha de generación de la clave privada.

## c. Generar una clave de revocación

Generar una clave de revocación le permitirá revocar su clave, o sea, cancelar su validez. Está firmada por su clave privada. Piense en dos casos: si le han robado su clave privada, hay que cancelarla para que deje de ser válida. Si ha perdido su passphrase, también tendrá que revocar su clave. En ambos casos, tendrá que volver a generar una clave. La clave de revocación permite a todas las personas que disponen de una clave pública comprobar que la cancelan. Una vez generada la clave de revocación, consérvela en un sitio seguro, no la publique ni tampoco la pierda.

Por lo tanto, es muy importante crear una clave de revocación justo después de haber creado el par de claves pública/privada y guardarla en un lugar seguro. Eso es así por varias razones: si usted pierde por ejemplo su clave privada, no tendrá ya la oportunidad de generar la clave de revocación. Si se olvida de su passphrase, no podrá utilizar más su clave privada; por lo tanto, no podrá crear la clave de revocación.

En contraposición, tenga cuidado en no divulgar su clave de revocación: si se conociera, cualquiera podría revocar una clave que usted utiliza.

Inicie el comando **gpg** con **--gen-revoke** y el nombre o el comentario (alias, seudo) asociado a la clave. Teclee la razón por la cual quiere generar una clave de revocación. Dé la razón que quiera y el comentario asociado con la elección, y confirme.

```
seb@slyserver:~> gpg --gen-revoke Slyce
sec 1024D/13E021A8 2009-05-04 Sébastien ROHAUT (Slyce)
<sebastien.rohaut@dominio.es>

¿Generar un certificado de revocación para esta clave? (s/N) s
elija la causa de la revocación:
 0 = Ninguna razón especificada
 1 = Se comprometió la clave
 2 = Se sustituyó la clave
 3 = No se utiliza más la clave
 0 = Cancelar
(Aquí seguramente tendría que seleccionar 1)
¿Su decisión? 1
```

Entre una descripción opcional; terminela con una línea vacía:

Luego, tiene que teclear, como cada vez que se debe utilizar la clave privada, su frase de contraseña:

Ahora la clave está generada. El sistema le muestra las consignas de uso: proteja su clave.

```
> Ejemplo público para libro
>
Causa de revocación: Se comprometió la clave.
Ejemplo público para libro.
¿Está de acuerdo? (s/N) s
```

Deje su clave GPG en un soporte seguro: imprímala, póngala en un cd grabado (incluso de forma exclusiva), en una clave USB (no confíe en este soporte), en una caja fuerte o aprendráselo de memoria (si es buena).

```
Necesita una frase de contraseña para desbloquear su clave secreta para el usuario. <Sébastien ROHAUT (Slyce)>
<sebastien.rohaut@dominio.es>>
clave de 1024 bits DSA, ID 13E021A8, creada el 2009-05-04
```

```
salida con formato ASCII forzado.
Certificado de revocación creado.
Muévalo a un soporte que pueda guardar; si Mallory tiene acceso a este certificado, lo puede utilizar para hacer inutilizable su clave.
Una buena idea consiste en imprimir este certificado y luego guardarlo en otro sitio, en caso de que el soporte se vuelva ilegible. Pero cuidado: el sistema de impresión de su máquina podría almacenar estos datos y hacerlos accesibles a otras personas!
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2.0.9 (GNU/Linux)
Comment: A revocation certificate should follow
```

```
iGIEIBECACIFAKoAQc8bHQJFeGVtcGxlIHBlYmxpYyBwb3VyIGxpdnJlAAoJEB8S
...
jlkHZg==9xGd
-----END PGP PUBLIC KEY BLOCK-----
```

## d. Gestionar el almacén de claves

Haga la lista de las claves generadas por GPG con el parámetro `--list-key`. Para cada clave, puede ver la información relativa a las claves públicas y privadas, su propietario y la posible fecha de expiración.

```
seb@slyserver:~> gpg --list-key  
/home/seb/.gnupg/pubring.gpg  
-----  
pub 1024D/13E021A8 2009-05-04  
uid Sébastien ROHAUT (Slyce) <sebastien.rohaut@dominio.es>  
sub 1024g/BA311C5C 2009-05-04  
...  
...
```

Se muestran las firmas, o más bien la lista de las firmas de cada una de sus claves, por clave pública y clave privada, con la `--list-sigs` que

incluye la `--list-key`. Más adelante

# Validación de los conocimientos adquiridos: preguntas/respuestas

## 1. Preguntas

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las preguntas siguientes:

### **Seguridad local**

- 1** ¿Cuáles son los tres principales objetivos de la seguridad informática?
- 2** ¿Por qué es importante controlar los derechos SUID y SGID de sus binarios?
- 3** Dé el comando que permite buscar y visualizar la información detallada relativa a los ficheros ordinarios en /usr cuyo propietario es root y que dispone del privilegio SUID.
- 4** Un fichero del paquete rpm bash le resulta raro. El resultado de la verificación RPM indica .M.....T /bin/bash. ¿Qué ha ocurrido?
- 5** ¿Es pertinente forzar a los usuarios a cambiar de contraseña cada semana, sabiendo que la contraseña debe responder a unas reglas estrictas?
- 6** ¿Cómo comprobar la información de contraseña de cualquier usuario?
  - A - chage -l user
  - B - passwd -l user
  - C - grep ^user /etc/shadow
  - D - passwd -S user
- 7** ¿Cuál es el inconveniente principal del funcionamiento por defecto de pwgen para los españoles?
- 8** Genere seis contraseñas aleatorias de una longitud de 8 caracteres.
- 9** El usuario ftponly no debe poder conectarse de manera interactiva. Por lo tanto, ¿qué comando puede usted teclear para modificar su shell de conexión?
- 10** ¿Cómo bloquear toda conexión nueva?
  - A - pasando en init 1.
  - B - modificando todos los shells en /bin/false.
  - C - bloqueando todos los terminales dentro de /etc/securetty.
  - D - creando el fichero /etc/nologin.
- 11** Un «scan» rápido por «John The Ripper» muestra que éste ha pasado hace apenas unos segundos en determinados logins. ¿Por qué?
- 12** ¿Cuál es el objetivo de un rootkit?
- 13** ¿Por qué debería considerarse la instalación de un antivirus en un servidor que dispone de particiones de ficheros?
- 14** ¿Qué línea de comandos tiene que teclear para actualizar la base de datos de las firmas de virus clamav?
- 15** ¿Sabía que la simple línea «:{ :|:& };» en bash puede provocar la parada de su máquina? Es una «fork bomb»: crea miles de procesos y termina por saturar la máquina. ¿Cómo puede impedir que sus usuarios creen demasiados procesos?
- 16** ¿Cómo conservar (en el boot) las configuraciones de ulimit?
  - A - creando un script lanzado por init.

- B - modificando /etc/security/limits.conf.
- C - modificando el fichero /etc/profile con una serie de ulimit.
- D - modificando el fichero ~/.bashrc de los usuarios y en /etc/skel.

**17** Un boletín de seguridad del CERT le informa de un fallo de seguridad de un producto instalado en su puesto. ¿Qué tiene que hacer?

### **Seguridad de la red**

**18** ¿Cómo aislar los puertos abiertos y las conexiones activas en su máquina?

**19** ¿Para qué sirve el comando **nmap**?

- A - Facilitar una tarjeta de la red.
- B - Probar los puertos remotos de una máquina de la red.
- C - Obtener información de seguridad en una máquina de la red.
- D - Conocer el sistema operativo de una máquina remota.

**20** ¿Cómo se puede emplear de manera malintencionada la información devuelta por nmap?

**21** El servicio smbd parece ejecutarse en su máquina y, sin embargo, usted no tiene particiones de Windows. ¿Qué tendría que hacer?

**22** ¿Cómo saber si un servicio integra los TCP Wrappers?

**23** ¿Cuál es la ruta a los dos ficheros de configuración de los TCP Wrappers?

**24** Prohíba la conexión al servicio sshd a todo el mundo salvo a la subred 10.17.32.0/22.

**25** Si no pudiera contar con NetFilter, ¿cómo prohibiría todo intento de conexión a su máquina?

**26** ¿Cómo suprimir todas las reglas de una sola vez?

**27** Prohíba toda conexión de la subred 10.17.32.0/22 en su máquina, salvo 10.17.32.5.

**28** ¿Cómo impedir cualquier conexión TCP saliente con destino a un servicio ssh (puerto 22)?

**29** Su PC dispone de dos tarjetas de red. ¿Cómo prohibir cualquier entrada o salida en la segunda?

**30** Su PC, que dispone de dos tarjetas de red, sirve de enrutador. Quiere impedir el encaminamiento del puerto TCP 22. ¿Cómo hacerlo?

## **2. Resultados**

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /30

Para este capítulo, su resultado mínimo debe ser de 22 respuestas acertadas (22/30).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- Seguridad local.
- Seguridad de la red.

## **3. Respuestas**

### **Seguridad local**

**1** ¿Cuáles son los tres principales objetivos de la seguridad informática?

*La seguridad de conexión, la integridad de los datos y la confidencialidad de los datos.*

**2** ¿Por qué es importante controlar los derechos SUID y SGID de sus binarios?

*Un comando con los derechos SUID asigna los derechos de su propietario al que lo utiliza. Si el comando es un shell o cualquier otro comando peligroso como cat, las consecuencias pueden ser desastrosas.*

**3** Dé el comando que permite buscar y visualizar la información detallada relativa a los ficheros ordinarios en /usr cuyo propietario es root y que dispone del privilegio SUID.

`find /usr -user root -type f -perm -4000 -ls`

**4** Un fichero del paquete rpm bash le resulta raro. El resultado de la verificación RPM indica .M.....T /bin/bash. ¿Qué ha ocurrido?

*M indica que se han modificado los derechos del fichero, y T que, la fecha de modificación también. Compruebe los derechos de /bin/bash y, si es necesario, vuelva a colocarlos correctamente (755).*

**5** ¿Es pertinente forzar a los usuarios a cambiar de contraseña cada semana, sabiendo que la contraseña debe responder a unas reglas estrictas?

*No, o más bien, no necesariamente. Al cambiar demasiado a menudo, los usuarios olvidarán la contraseña o la apuntarán. Obtendrá el efecto inverso de lo que desea.*

**6** ¿Cómo comprobar la información de contraseña de cualquier usuario?

- A - chage -l user
- B - passwd -l user
- C - grep ^user /etc/shadow
- D - passwd -S user

*A y D. El resultado de A es más legible.*

**7** ¿Cuál es el inconveniente principal del funcionamiento por defecto de pwgen para los españoles?

*Da los resultados en inglés, menos comprensibles para nosotros.*

**8** Genere seis contraseñas aleatorias de una longitud de 8 caracteres.

`pwgen -s 8 6`

**9** El usuario ftponly no debe poder conectarse de manera interactiva. Por lo tanto, ¿qué comando puede usted teclear para modificar su shell de conexión?

`usermod -s /bin/false ftponly`

**10** ¿Cómo bloquear toda conexión nueva?

- A - pasando en init 1.
- B - modificando todos los shells en /bin/false.
- C - bloqueando todos los terminales dentro de /etc/securetty.
- D - creando el fichero /etc/nologin.

*D. El módulo pam\_nologin prohibirá cualquier conexión nueva.*

**11** Un «scan» rápido por «John The Ripper» muestra que éste ha pasado hace apenas unos segundos en determinados logins. ¿Por qué?

*La contraseña es obvia, es probable que esté relacionada con el nombre del propio login o con una variante. Fuerce a estos usuarios a cambiar su contraseña.*

**12** ¿Cuál es el objetivo de un rootkit?

*El objetivo de un pirata es obtener el mayor número de privilegios posibles en una máquina. Se instala un rootkit en local vía cualquier fallo (humano o de software) del sistema con el objetivo de abrir una puerta secreta mucho más accesible.*

**13** ¿Por qué debería considerarse la instalación de un antivirus en un servidor que dispone de particiones de ficheros?

*Porque algunos ficheros que contengan virus pueden circular a sus espaldas sobre las diversas particiones, sobre todo si sus usuarios están con Windows. Los virus específicos de Linux son muy raros.*

**14** ¿Qué línea de comandos tiene que teclear para actualizar la base de datos de las firmas de virus clamav?

*Teclee `freshclam` simplemente.*

**15** ¿Sabía que la simple línea «`:(){ :|:& };:`» en bash puede provocar la parada de su máquina? Es una «fork bomb»: crea miles de procesos y termina por saturar la máquina. ¿Cómo puede impedir que sus usuarios creen demasiados procesos?

*ulimit permite modificar el número de proceso por usuario con el parámetro `-u`; `ulimit -u 1024` coloca el límite a 1024 procesos por usuario.*

**16** ¿Cómo conservar (en el boot) las configuraciones de ulimit?

- A - creando un script iniciado por init.
- B - modificando /etc/security/limits.conf.
- C - modificando el fichero /etc/profile con una serie de ulimit.
- D - modificando el fichero ~/.bashrc de los usuarios y en /etc/skel.

*B. Son los módulos PAM los que se encargarán de modificar el entorno del usuario a su conexión.*

**17** Un boletín de seguridad del CERT le informa de un fallo de seguridad de un producto instalado en su puesto. ¿Qué tiene que hacer?

*Compruebe en el sitio de su distribución si la alerta le corresponde. Si es el caso, actualice su distribución.*

## **Seguridad de la red**

**18** ¿Cómo aislar los puertos abiertos y las conexiones activas en su máquina?

*Con el comando netstat ya tratado: `netstat -a -A inet`*

**19** ¿Para qué sirve el comando nmap?

- A - Facilitar una tarjeta de la red.
- B - Probar los puertos remotos de una máquina de la red.
- C - Obtener información de seguridad en una máquina de la red.
- D - Conocer el sistema operativo de una máquina remota.

*B, C y D. nmap es una verdadera navaja suiza de la seguridad de red.*

**20** ¿Cómo se puede emplear de manera malintencionada la información devuelta por nmap?

Además de la presencia de numerosos puertos abiertos, nmap intenta detectar el nombre del servicio asociado y su versión. De ahí, vía los boletines de seguridad, se puede intentar encontrar y luego aprovechar un fallo. Si un servicio no está encriptado, por ejemplo telnet, es más sencillo: basta con colocar un «sniffer» (wireshark, por ejemplo) para visualizar el tráfico de forma transparente.

**21** El servicio smbd parece ejecutarse en su máquina y, sin embargo, usted no tiene particiones de Windows. ¿Qué tendría que hacer?

Suprime este servicio al arrancar y párelo. De manera general, suprime todos los servicios inútiles y reactívelos en caso de necesidad.

**22** ¿Cómo saber si un servicio integra los TCP Wrappers?

Se compila la librería de manera estática dentro del binario. Mire la cadena «host\_access» en el programa con el comando strings.

**23** ¿Cuál es la ruta a los dos ficheros de configuración de los TCP Wrappers?

/etc/hosts.allow y /etc/hosts.deny.

**24** Prohíba la conexión al servicio sshd a todo el mundo salvo a la subred 10.17.32.0/22.

En /etc/hosts.deny escriba sshd :ALL. En /etc/hosts.allow escriba sshd :10.17.32.0/255.255.252.0.

**25** Si no pudiera contar con NetFilter, ¿cómo prohibiría todo intento de conexión a su máquina?

Modifique la regla por defecto en entrada: iptables -P INPUT DROP

**26** ¿Cómo suprimir todas las reglas de una sola vez?

Con el parámetro -F: iptables -F

**27** Prohíba toda conexión de la subred 10.17.32.0/22 en su máquina, salvo 10.17.32.5.

iptables -A INPUT -s 10.17.32.0/22 ! 10.17.32.5 -j DROP

**28** ¿Cómo impedir cualquier conexión TCP saliente con destino a un servicio ssh (puerto 22)?

iptables -A OUTPUT -p tcp -dport 22

**29** Su PC dispone de dos tarjetas de red. ¿Cómo prohibir cualquier entrada o salida en la segunda?

Con dos reglas iptables: iptables -A INPUT -i eth0 -j DROP. Para la segunda, sustituya INPUT por OUTPUT.

**30** Su PC, que dispone de dos tarjetas de red, sirve de enrutador. Quiere impedir el encaminamiento del puerto TCP 22. ¿Cómo hacerlo?

iptables -A FORWARD -i eth0 -o eth1 -p tcp -dport 22 -j DROP

# Prácticas

## 1. Control de los ficheros

Objetivo: controlar los ficheros y derechos asociados.

- Para cada fichero, es posible establecer una suma de control o checksum. El comando **md5sum** calcula la suma de control de un fichero en formato MD5. Calcule el checksum del fichero /etc/passwd:

```
# md5sum /etc/passwd  
8737073e74b3035af61911a6a4323b53 /etc/passwd
```

- Modifique una información, aunque sea sencilla, por ejemplo un comentario, en /etc/passwd, y vuelva a calcular el MD5. Aquí se ha sustituido una «é» por una «e»:

```
$ md5sum /etc/passwd  
a7450bcc205cbee80c429d9522c43f53 /etc/passwd
```

La suma es totalmente diferente, lo que muestra que se ha modificado el fichero. Si conserva en un fichero MD5SUM las sumas de control de origen, podrá detectar las modificaciones de manera más fiable que con la fecha.

- Durante la búsqueda siguiente:

```
# find ! -user root -perm -4000
```

Encuentra un programa /usr/bin/lppasswd particular:

```
-rwsr-xr-x 1 lp      sys        14112 abr 18 00:55  
.bin/lppasswd
```

El derecho SUID está activado y el propietario no es root. ¿Por qué?

El propietario de los ficheros manejados por **Ippasswd** es el usuario **lp**. El uso del SUID-bit no depende del propietario root. En este caso, se ejecutará el comando **Ippasswd** como usuario **lp**.

## 2. Seguridad de los usuarios

Objetivo: gestionar la política de seguridad de las contraseñas de los usuarios.

- Supongamos que actualmente, no existe ninguna política de seguridad de los usuarios, son libres de modificar su contraseña a su antojo. Usted ha utilizado el comando **chage** para modificar la política de seguridad para los usuarios existentes. Ahora va a modificar esta política para todos los futuros usuarios. Modifique el fichero /etc/login.defs para definir un cambio de contraseña cada 40 días, la prohibición de cambiar la contraseña antes de 7 días, y que se avise al usuario 10 días antes:

```
PASS_MAX_DAYS 40  
PASS_MIN_DAYS 7  
PASS_WARN_AGE 10
```

- Como root, quiere modificar sobre la marcha las contraseñas de todos los usuarios usando un script. Se genera la contraseña de 8 caracteres con pwck. Los UID de los usuarios empiezan en 1000. Construya primero un bucle que lea el fichero /etc/passwd:

```
while read line  
do
```

```
...  
done </etc/passwd
```

3. Para cada línea, aísle el login y el UID:

```
user=$(echo $linea| cut -d: -f1)  
uid=$(echo $linea| cut -d: -f3)
```

4. Genere una contraseña segura con 8 caracteres y colóquela en una variable llamada `pass`:

```
pass=$(pwgen -s -1)
```

5. Modifique la contraseña del usuario:

```
echo $pass | passwd --stdin $user
```

El script completo es:

```
while read line  
do  
user=$(echo $linea| cut -d: -f1)  
uid=$(echo $linea| cut -d: -f3)  
pass=$(pwgen -s -1)  
echo $pass | passwd --stdin $user  
done </etc/passwd
```

### 3. Seguridad general del sistema

Objetivo: evitar los rootkits, virus y controlar los límites.

1. El comando **chkrootkit** permite la detección de los rootkits más corrientes en un sistema. Usted decide colocar el comando en crontab. Se ejecutará cada día a la 1.00h de la madrugada. Cree un fichero cron\_rootkit en /etc/cron.d y coloque en él la línea siguiente:

```
0 1 * * * /sbin/chkrootkit >/tmp/rootkit
```

2. Los resultados se colocan en /tmp/rootkit, pero no es una buena idea. Lo mejor sería recibirlos por mail:

```
0 1 * * * /sbin/chkrootkit | mail user@server -s "Resultados rootkit  
$(date)"
```

3. Efectúe una actualización de la base de los antivirus de clamav cada dos días a las 2.00h de la madrugada, según el mismo principio:

```
0 2 */2 * * freshclam >/dev/null 2>&1
```

4. Sus usuarios suelen consumir demasiados recursos. Limite a las personas del grupo users a 256 procesos. Modifique el fichero `/etc/security/limits.conf` añadiendo la línea siguiente:

```
@users hard nproc 256
```

5. Sea cual sea la distribución, intente siempre actualizar lo más a menudo posible sus paquetes por una cuestión de seguridad:

Debian y Ubuntu: apt-get upgrade  
Red Hat y Fedora: yum update  
openSUSE: zypper update

## 4. Seguridad de red

Objetivo: comprobar los puertos, la configuración del cortafuegos y de los TCP Wrappers.

1. Inicie nmap en su propia máquina. Compare los resultados con los de netstat -A inet -a.  
¿Cuál es la principal diferencia?  
Nmap escanea los puertos abiertos en su máquina, no las conexiones salientes. netstat da la lista de los puertos locales y remotos abiertos, así como de los procesos asociados.  
Pero, sobre todo, nmap facilita, cuando los puede determinar, el nombre real y las versiones de los servicios y del sistema operativo probados. Nmap es tanto una herramienta de seguridad como de hacking.
2. Un producto como Wireshark (ex Ethereal) permite «esnifar» una red: está a la escucha de todo el tráfico, puede grabarlo, filtrarlo, etc. Dos servicios se ejecutan en un servidor: telnetd y sshd. Si escucha el tráfico de y hacia este servidor, debería notar algo. ¿Qué?  
El tráfico a destino y desde el puerto 22 es ilegible: está encriptado. No hay manera sencilla de recuperar o analizar el contenido. Por el contrario, el tráfico del puerto 23 es totalmente legible: no se ha añadido seguridad a la comunicación. Todo pasa sin codificar, incluyendo las contraseñas. ¿Qué tiene que hacer?  
Desactive el servicio telnetd.
3. Puede elegir entre los TCP Wrappers para colocar una protección de tipo cortafuegos. ¿Qué principio debe guiar su solución?  
El cortafuegos netfilter trabaja a nivel de los protocolos: filtra las direcciones, los puertos, los protocolos. El control de acceso se hace al nivel del núcleo.  
Los TCP Wrappers están orientados, como indica su nombre, TCP; por lo tanto, actúan en la capa de transporte y los servicios binarios: el control de acceso se hace al nivel del servicio.
4. El fichero `/etc/hosts.allow` contiene:

`sshd: ALL`

El fichero `/etc/hosts.deny` contiene:

`sshd: ALL EXCEPT 192.168.1.25`

Puede comprobar que todo el mundo se conecta vía ssh. Es evidente que existe un error en la configuración: sólo 192.168.1.25 debería lograrlo. Corrija.

Tiene varias soluciones:

- Suprima la línea de hosts.allow y deje hosts.deny intacta.
- Modifique hosts.allow:

`sshd: 192.168.1.25`

Y modifique hosts.deny de la manera siguiente:

`sshd: ALL`

5. Cree reglas netfilter que prohíban a los usuarios cualquier conexión tcp en el puerto 23, salvo para las máquinas de la subred 192.168.1.0/24:

```
# iptables -A INPUT -p tcp --dport 23 -s ! 192.168.1.0/24 -j DROP
```

# Requisitos y objetivos

## 1. Requisitos

- Disponer de hardware compatible con el servidor X.
- Tener acceso como root.
- Saber utilizar un editor.
- Disponer de una o varias cuentas de prueba.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Entender el funcionamiento de X Window.
- Entender los fundamentos de los gestores de ventanas.
- Manejar los principales toolkits y widgets.
- Instalar xorg.
- Configurar xorg mediante los comandos estándares.
- Comprender y modificar a mano el fichero xorg.conf.
- Acceder al registro del servidor X.
- Configurar los gestores de visualización xdm, gdm, kdm.
- Iniciar el sistema en modo gráfico.
- Abrir una sesión.
- Configurar su sesión personal.
- Elegir un gestor de ventana o un entorno de escritorio.
- Exportar la visualización hacia otro servidor X.
- Modificar los recursos de los programas X.

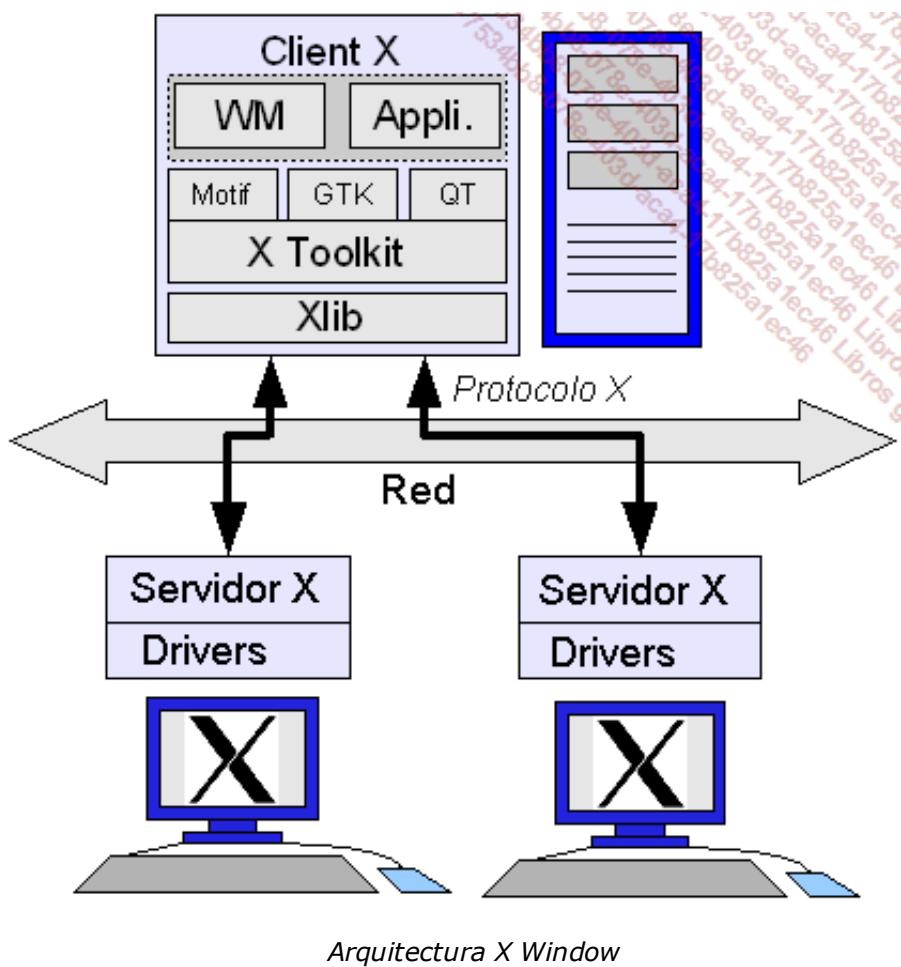
# ¿Cómo funciona un entorno gráfico?

## 1. X Window System

### a. Un modelo cliente/servidor

Mientras en otros sistemas operativos se integra la interfaz gráfica en lo más profundo del sistema, Unix y Linux disponen de una arquitectura gráfica totalmente diferente. El sistema gráfico básico se llama **X Window System** o, de manera más corriente, **X Window**, **X11** o sencillamente **X**.

X no es sólo un simple programa. Es un sistema gráfico completo encargado de dibujar y gestionar los eventos de los componentes habituales de un entorno gráfico de usuario **GUI** (*Graphical User Interface*): ventanas, botones, menús, listas, menús desplegables, casillas de selección, cursores de ratón, etc. Tome nota: X puede gestionar y mostrar estos componentes gráficos, pero no se encarga de instalarlos. X sólo gestiona las interacciones entre el hombre y la máquina.



X tiene otra particularidad: es cliente/servidor. A menudo, el servidor X es un componente de software en un ordenador que dispone de un teclado, un ratón y una pantalla. Recibe y responde a órdenes de visualización, o procedentes del teclado o ratón. El cliente X se conecta al servidor y le manda órdenes de visualización, peticiones de entrada vía teclado o el estado del ratón. Dicho de otro modo, un cliente X es un programa capaz de dialogar con el servidor X. En realidad, un cliente X es un programa gráfico. Para poder comunicarse con el servidor, utiliza un componente llamado Xlib. El cliente y el servidor no siempre están en la misma máquina. El servidor que gestiona la visualización puede encontrarse en un primer ordenador, y el programa gráfico en otro. Las órdenes, llamadas peticiones, entre el cliente y el servidor pasan por la red.

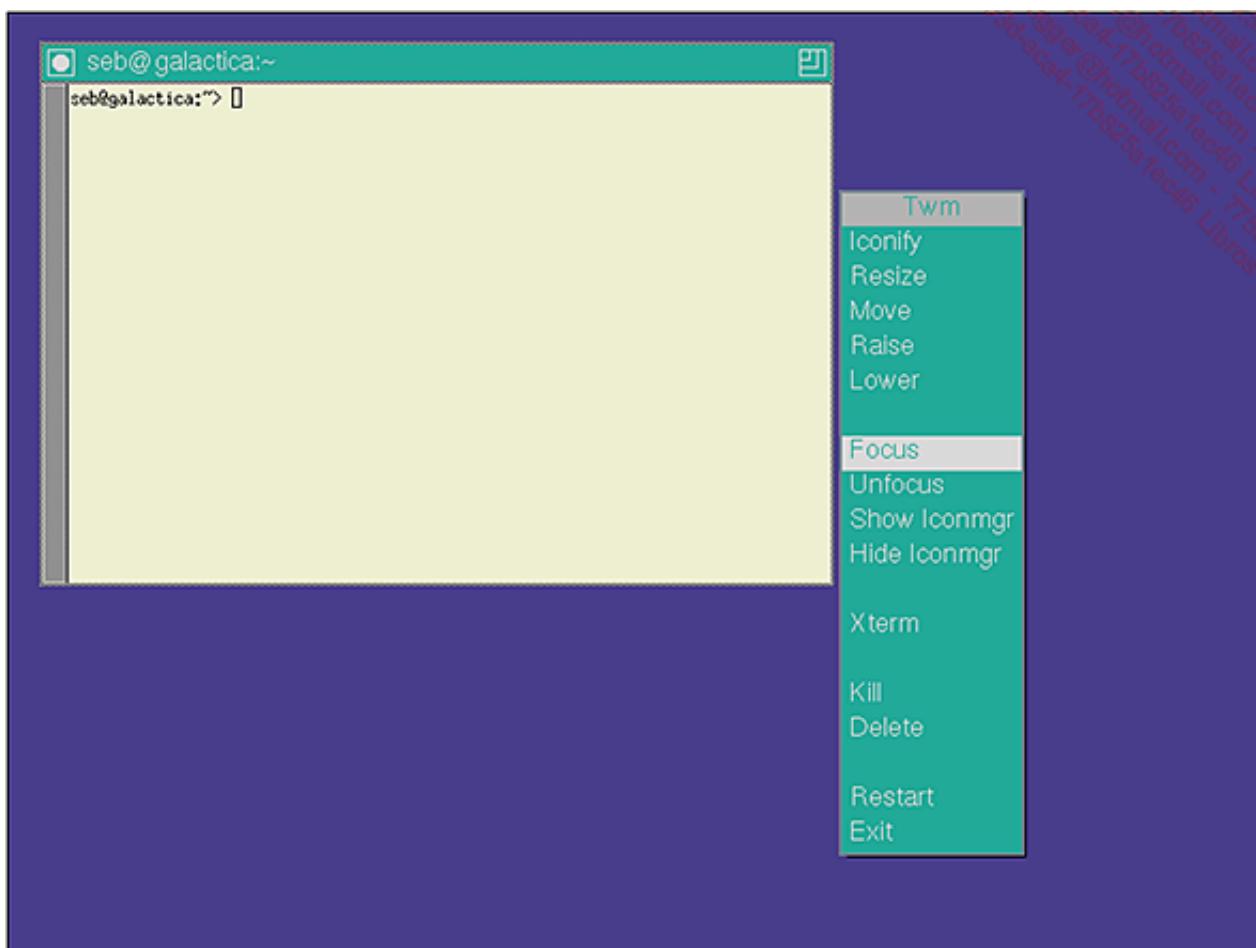
- 
- No confunda X Window con Windows. El primero es un sistema de visualización cliente/servidor; el segundo, un sistema operativo. El «Window» de X Window no tiene «s». Además, Windows y X Window no son compatibles entre sí, aunque sea posible instalar un servidor X en Windows.
- 

Un usuario puede iniciar un programa (cliente) en su máquina local conectada por la red a un servidor remoto donde se está ejecutando realmente el servidor de ventanas.

Si usted ejecuta solo X Window, ejecuta únicamente el servidor. El resultado puede sorprenderle: una pantalla gris con una cruz como cursor de ratón. Por mucho que lo intente, no habrá ventana que venga a alegrarle la vista y los clics del ratón no tienen efecto alguno. Aunque iniciara un cliente, notaría que no hay ventanas.

## b. El gestor de ventanas

Entre las muchas peticiones gráficas, la más importante es la de crear una ventana y decorarla dibujando los diversos elementos de ésta: la barra de título, el marco, los diversos botones. Como X sólo facilita lo imprescindible, se limita a dibujar la ventana, pero no le corresponde determinar cómo se debe renderizar el resto de los elementos. Otro programa cliente X debe decir al servidor cómo dibujar la ventana: es el **gestor de ventanas** o *Window Manager*. El servidor X visualiza el resultado dibujado por este gestor: ventanas, selecciones, movimientos y decoraciones (estilos, colores, etc.).

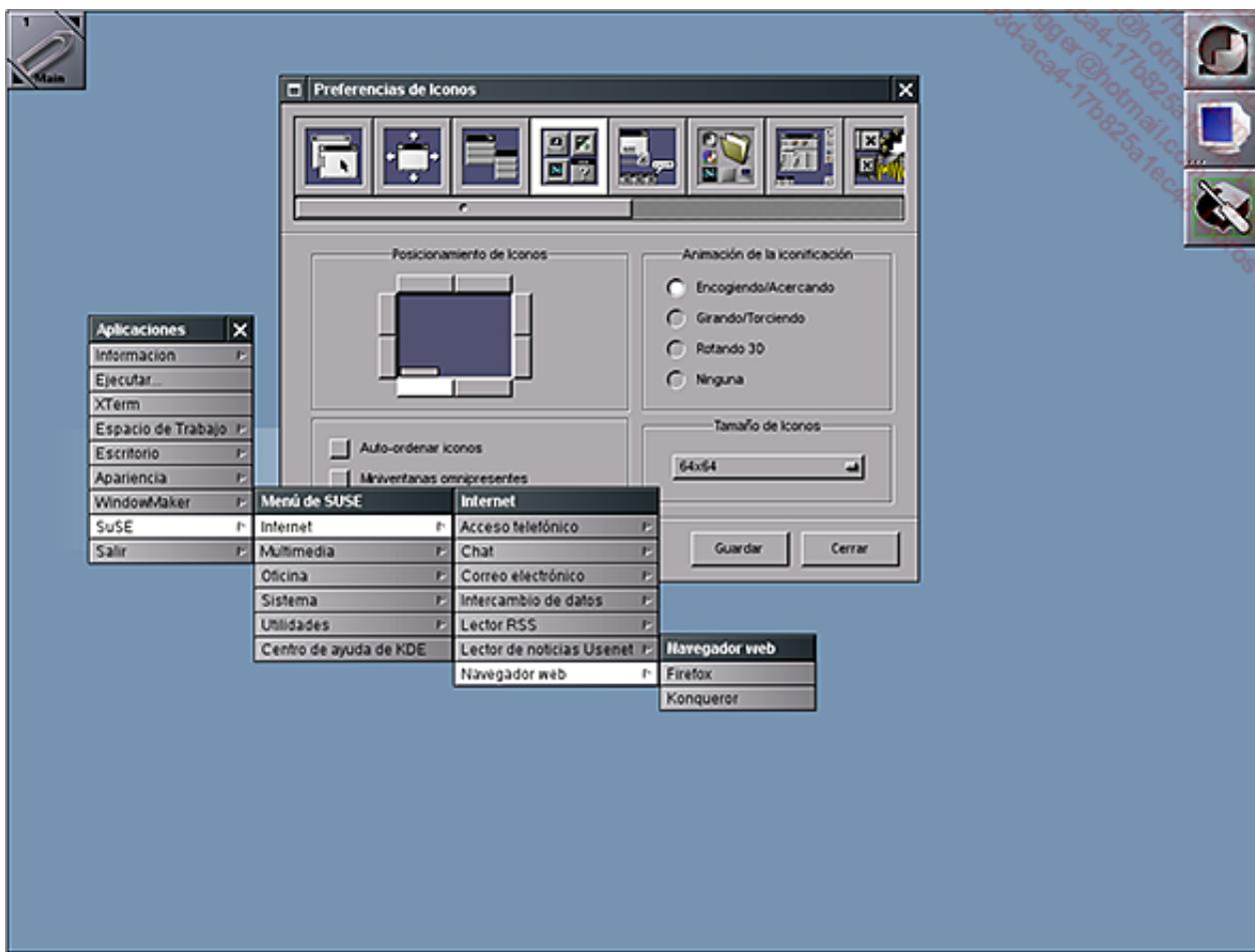


Un gestor simple: twm

Esto quiere decir que no hay un único gestor de ventanas, sino varios. Algunos son muy sencillos y básicos, y se limitan a lo mínimo; por ejemplo TWM, que aquí mostramos con una ventana y un menú. Observe su austeridad espartana. Sin duda, no es la mejor manera de sorprender a sus amigos con la calidad gráfica de

Linux.

Otros son muy completos y permiten trabajar muy cómodamente porque, además de las ventanas básicas, proponen temas visuales agradables y personalizables, menús contextuales y, a veces, paneles de configuración. Éste es el caso de WindowMaker, con el que es posible incluso cambiar menús, temas visuales, añadir botones en la barra de la derecha, etc. Durante mucho tiempo, este gestor de ventanas fue el más utilizado: rápido, eficaz, frugal en el consumo de recursos.



WindowMaker, un window manager evolucionado

## c. Los widgets y los toolkits

### Los componentes y su librería

Incluso a los gestores de ventanas más desarrollados les falta algo: una integración más avanzada de los programas (y de sus estilos) en el entorno. La razón estriba en el hecho de que, salvo casos muy especiales, los **WM** (*Windows Manager*) usan librerías gráficas propias con funcionalidades y componentes específicos. A los elementos que forman parte de la interfaz gráfica (botón, menú, campo de entrada, etc.) se les conoce como **Widget** (*Window gadget*), o sea, «chisme» para ventanas. Las librerías gráficas contienen juegos completos.

Todas estas librerías gráficas deben proporcionar funciones para dibujar y gestionar widgets. Cuando la librería contiene un kit completo y extendido de widget, se llama **Widget Toolkit**, o de manera más sencilla **toolkit** o caja de herramientas para la interfaz gráfica. Existen varios, que se diferencian según su complejidad, su uso, su estética (relativamente), el lenguaje de programación para utilizarlos, etc. X Window dispone de un toolkit por defecto llamado **Xt** (*X Toolkit*). El más conocido y utilizado ha sido durante mucho tiempo **MOTIF**, más aún porque es un estándar POSIX IEEE, hasta que, liberado en torno al año 2000,

MOTIF fue considerado software propietario y, por esa razón, los programadores gráficos crearon otros conjuntos de componentes. La consecuencia es de lo más interesante: una pléyade de toolkits diferentes, incompatibles entre sí, que dibujan de manera totalmente diferente los widgets. Todavía puede encontrar en Linux muchos ejemplos en los que un componente aparece visualmente diferente en varios programas, dando la impresión de caos de estilos heterogéneos.

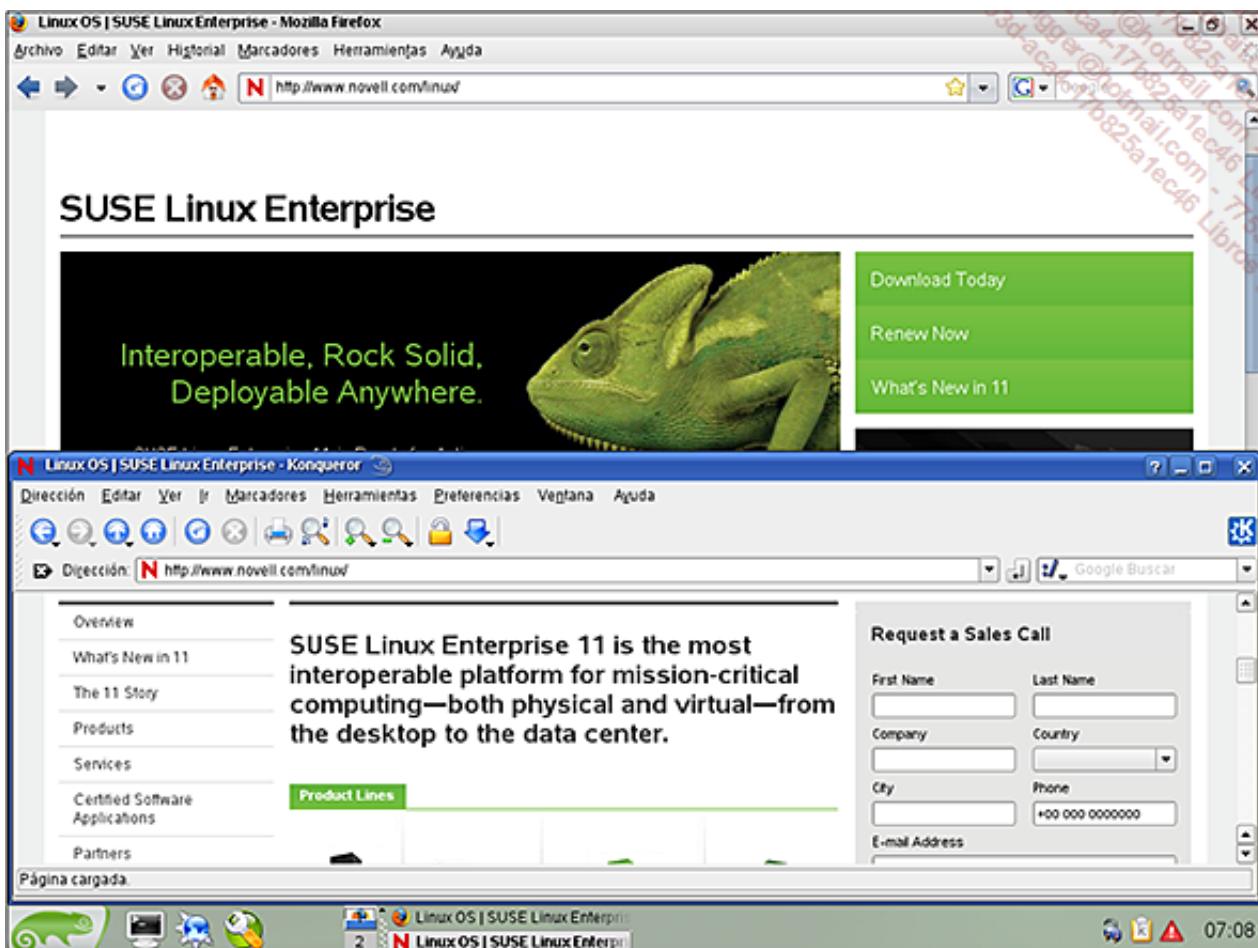
## **GTK, Qt**

Al no poder utilizar Motif (y su equivalente libre Lesstif, que no estaba todavía muy preparado), los desarrolladores de interfaces gráficas en Linux concentraron sus esfuerzos en dos toolkits concretos, excelentemente programados y que permiten hacer abstracción totalmente de X Window. Existen varias versiones que no necesitan siquiera X Window (para Windows, Macintosh, los asistentes personales, los teléfonos, etc.).

- El primero se llama **GTK** (*GIMP toolkit*). Fue creado para dibujar la interfaz gráfica del programa de retoque de imágenes GIMP. Gracias a su eficacia gráfica, muchos programadores lo han incorporado a sus propios programas y GTK con el tiempo se ha vuelto independiente. Hoy es la librería por defecto del entorno de escritorio GNOME. GTK está escrito en lenguaje C.
- El segundo se llama **QT** (*cute, mono*). La Q no significa nada: los desarrolladores de este toolkit pensaban que esta consonante, Q, era muy bonita en el editor de textos que utilizaban... Está desarrollado en C++, pero es compatible con una decena de lenguajes. Además de las funciones de dibujo y gestión de los widgets, QT propone un entorno completo de desarrollo de aplicaciones gráficas y no gráficas: bases de datos SQL, XML, multithreading, gestión de ficheros, internacionalización, etc. La librería QT es la utilizada por el entorno ofimático KDE.
- Hay otros muchos toolkits, como **MOTIF, AWT, ATHENA**. Aunque son menos conocidos, todavía hay aplicaciones que los utilizan.

GTK y QT no son compatibles, al menos de manera nativa. Si ejecuta un programa desarrollado con GTK en KDE, el contorno de la ventana tendrá el estilo de KDE (QT), pero GTK dibujará su contenido. Mire el resultado en la captura de pantalla siguiente. La ventana de arriba representa el navegador de Internet Firefox, que utiliza la librería GTK, mientras que la ventana de abajo representa el navegador Konqueror, que utiliza la librería QT. Los menús, los botones y el contenido no son idénticos. Afortunadamente, SUSE Linux hace bien las cosas y propone una homogeneización de los temas gráficos. En su parte final, este capítulo le explica cómo optimizar la visualización de sus temas gráficos.

Algunos programas utilizan sus propios toolkits. Es el caso de OpenOffice.org y es la principal razón de su lentitud durante su primera descarga. Cuando ejecuta este programa, éste debe cargar también su toolkit, lo cual resulta redundante con el de su gestor de ventanas o de su entorno ofimático. Afortunadamente, las últimas versiones se adaptan a su entorno, pero algunos programas (o más bien sus programadores) aún se resisten.



*Dos toolkits, dos estilos*

#### d. Los escritorios virtuales

Si hay una característica de Linux que se echa de menos en sus competidores, es la del escritorio virtual. ¿Qué hacer cuando hay tantas ventanas abiertas que estamos obligados a minimizarlas para recuperar las escondidas, o volver a dimensionarlas para verlas? Pues utilizar los escritorios virtuales.

X Window permite la utilización de varios espacios de trabajo en una misma sesión. En vez de tener un escritorio, puede tener dos, tres, cuatro... En efecto, puede en teoría obtener tantos escritorios como desee si dispone de bastante memoria. Es el gestor de ventanas, o el entorno de escritorio, el que elige (y por lo tanto, usted). KDE permite, por ejemplo, disponer de veinte escritorios.

El interés es obvio. En vez de concentrar todas sus ventanas en un solo escritorio, repártalas en los escritorios disponibles. Más aún: sólo precisa de un simple clic para ir de un escritorio a otro, o incluso para desplazar una ventana de un escritorio a otro. Además, es posible hacer que una ventana se siga visualizando en varios escritorios virtuales (es decir, «colgarla» en la pantalla).

## 2. Los entornos de escritorio

Si Unix y Linux en particular han conservado durante mucho tiempo la fama de sistema operativo para profesionales, universitarios, informáticos o excéntricos gurús, es porque faltaba un entorno gráfico orientado a ofimática y de fácil manejo. Ya en los primeros capítulos se comentaba que los hábitos que los usuarios han ido adquiriendo con las interfaces de Windows o MacOS no pueden desaparecer de un día a para otro. Durante años, los gestores de ventanas fueron difíciles de configurar. ¿Se le puede pedir a alguien acostumbrado al

ratón que modifique un fichero de configuración en modo texto, cuya sintaxis no es precisamente sencilla?

Hacia 1996, Linux alcanzó la madurez entre las empresas, pero sigue habiendo un mercado por conquistar. En el otoño de 1995, Windows puso un contundente final a la carrera de la mayoría de los sistemas operativos para PC de oficina. Los DR-DOS, PC-DOS y sobre todo OS/2, muy superiores, han sobrevivido primero y, finalmente, desaparecido. Los entornos gráficos de Unix y Linux no pueden competir en términos de facilidad de manejo y oferta de programas para el gran público con los sistemas de Microsoft y de Apple. Todo queda por hacer: hay que alcanzar a los competidores, mejorar e innovar.

Este titánico trabajo no desanima a los programadores. Ya que Linux, para el gran público, no es de fácil manejo, hay que crear un entorno ofimático gráfico adaptado a sus necesidades. Con pocos meses de diferencia dos equipos distintos asumen el proyecto.

## **KDE**



*Logo de KDE*

En octubre de 1996 un primer equipo liderado por Matthias Ettrich anuncia el lanzamiento del proyecto**KDE** (*K Desktop Environment*). La K ya no tiene significado particular (salvo al principio: *kool*) y sólo es la letra más cercana a la L de Linux en el alfabeto. KDE ofrece una interfaz gráfica y aplicaciones unificadas alrededor de un toolkit llamado QT y desarrolladas en C++. KDE 1 ya era impresionante en 1997 y permitía utilizar Linux como si fuera una mezcla de Windows 3.1 y 95. Pero fue a partir de KDE 2, ya en el año 2000, y gracias a una nueva arquitectura, cuando KDE reveló su potencia e igualó a los entornos gráficos de sus competidores. La salida de la versión 3 (la versión 3.5.9 data de febrero de 2008) va aún más lejos y, superando a todos sus competidores, integra los diversos componentes de forma avanzada. KDE es el entorno gráfico favorito de la mayoría de los usuarios de Linux; pretende ser el más completo en términos de integración, de configuración y de oferta de software (a riesgo de tener numerosos duplicados y dejar la opción al usuario). KDE también es el entorno de escritorio utilizado por Linus Torvalds, el creador de Linux.

La versión 4.0, publicada a principios de 2008, supone una remodelación total del entorno ofimático KDE. A diferencia de la evolución constante y continua entre las versiones 1.0 y 3.x, KDE 4 representa una ruptura total en relación con sus antecesores. La principal novedad son componentes llamados «plasmoides». De hecho, la ruptura es tan total que la versión 4.0 considerada como estable no lo es tanto en el momento de escribir este capítulo. La versión 4.1, anunciada en julio de 2008, es la primera que se puede utilizar en un entorno de producción. La versión 4.7.0 salió en julio de 2011.

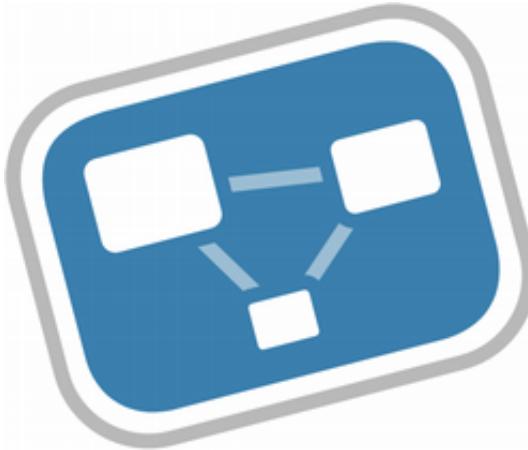
## **GNOME**



*Logo de Gnome*

En agosto de 1997, un equipo liderado por Miguel de Icaza y Federico Mena decide crear un entorno de escritorio totalmente libre para competir con KDE (cuyo QT no era libre en aquel momento). **GNOME** (*Gnu Network Object Model Environment*) es un entorno de escritorio basado en el toolkit GTK y programado en C. Es el entorno gráfico de escritorio oficial del proyecto GNU. La filosofía de GNOME difiere de manera radical de la de KDE. GNOME tiene un estilo voluntariamente depurado, que se centra en las funcionalidades esenciales de un entorno y que privilegia algunas aplicaciones en detrimento de otras. La integración no es tan exhaustiva, ya que las aplicaciones desarrolladas en GTK no son todas aplicaciones GNOME. GNOME 3, un remake completo, salió en abril de 2011.

### **El proyecto Freedesktop**



*Logo de Freedesktop*

Los desarrolladores de KDE y GNOME decidieron trabajar en una base común de configuración para una mejor integración de las aplicaciones GNOME en KDE, y viceversa, con el fin de armonizar la infraestructura del conjunto. El objetivo no es concebir un escritorio único, ya que cada uno tiene sus ventajas. El resultado es la creación de **Freedesktop**, una zona de comunicación y colaboración informales destinada a trabajar en favor de la interoperatividad de los diversos entornos gráficos para Linux y Unix. Freedesktop es neutral a nivel tecnológico, no se privilegia a ningún entorno. Pero, a menudo, lo que sale de Freedesktop se integra a GNOME y KDE: del menú principal común al programa de detección de nuevo hardware, pasando por los copiar y pegar, y los atajos de teclado y una multitud de mejoras que Freedesktop ha aportado.

# Xorg

## 1. Presentación

Desde su aparición hasta 2004, Linux (así como las distribuciones BSD) iba acompañado del entorno X Window libre XFree86. Un cambio de licencia que hizo que XFree86 fuera un poco menos libre y en particular que dejara de ser compatible con la GPL de la Free Software Foundation, tuvo como consecuencia un «fork» (creación de una nueva rama de desarrollo) a partir de la última versión con licencia MIT.

X.org termina por agrupar la mayoría de los antiguos desarrolladores de XFree86. X.org era un fork de XFree86 con fecha anterior al cambio de licencia, pero con ideas innovadoras bastante rechazadas por los partidarios de XFree86. X.org, llamado de manera común Xorg, se convierte en el entorno X Window principal de Linux y aún hoy lo sigue siendo.

X.org es rápido, modulable, está disponible con numerosos drivers, compatible. Las últimas versiones son capaces de la autodetección del hardware sin necesidad de complejos ficheros de configuración.



*Logo de Xorg*

La última versión de Xorg es la 7.3, y la 7.4 está próxima.

## 2. Instalación

Se entrega Xorg con todas las distribuciones de Linux. Se suelen suministrar las distribuciones anteriores a 2011 con Xorg 7.6. Resulta que el desarrollo y las innovaciones son tan rápidas que a veces les cuesta seguir el ritmo a las herramientas asociadas (en particular para la configuración).

Puede recuperar el código fuente de X.org, pero debe saber que le espera horas y horas de compilación. Preferirá instalar Xorg desde los CD, DVD o repositorios de su distribución.

Por defecto, se instala Xorg en /usr/X11R6. Sin embargo, algunas distribuciones colocan ahora los binarios, las librerías, los ficheros compartidos, etc., en el árbol clásico /usr y en particular los módulos y drivers en /usr/lib/xorg (o /usr/lib64/xorg).

En cambio, la configuración se encuentra siempre en el mismo sitio: /etc/X11.

A continuación, le presentamos un ejemplo de paquetes de Xorg instalados en una distribución Ubuntu 11.04 de 64 bits:

```
# dpkg -l "*xorg*" | grep ^ii|awk '{print $2}'  
xorg  
xorg-docs-core  
xserver-xorg  
xserver-xorg-core  
xserver-xorg-input-all  
xserver-xorg-input-evdev  
xserver-xorg-input-mouse  
xserver-xorg-input-synaptics  
xserver-xorg-input-vmmouse  
xserver-xorg-input-wacom  
xserver-xorg-video-all  
xserver-xorg-video-apm  
xserver-xorg-video-ark  
xserver-xorg-video-ati  
xserver-xorg-video-chips  
xserver-xorg-video-cirrus  
xserver-xorg-video-fbdev  
xserver-xorg-video-i128  
xserver-xorg-video-intel  
xserver-xorg-video-mach64  
xserver-xorg-video-mga  
xserver-xorg-video-neomagic  
xserver-xorg-video-nouveau  
xserver-xorg-video-openchrome  
xserver-xorg-video-qxl  
xserver-xorg-video-r128  
xserver-xorg-video-radeon  
xserver-xorg-video-rendition  
xserver-xorg-video-s3  
xserver-xorg-video-s3virge  
xserver-xorg-video-savage  
xserver-xorg-video-siliconmotion  
xserver-xorg-video-sis  
xserver-xorg-video-sisusb  
xserver-xorg-video-tdfx  
xserver-xorg-video-trident  
xserver-xorg-video-tseng  
xserver-xorg-video-vesa  
xserver-xorg-video-vmware  
xserver-xorg-video-voodoo
```

Observe aquí la presencia de numerosos paquetes de 32 bits. Recuerde que es posible tener a la vez librerías en 32 y 64 bits, siempre que se encuentren en rutas diferentes. Un programa de 32 bits sólo funciona con librerías de 32 bits; un programa de 64 bits, con librerías de 64 bits.

```
$ rpm -ql xorg-x11-libXv-32bit-7.2-61  
/usr/lib/libXv.so.1  
/usr/lib/libXv.so.1.0.0  
$ rpm -ql xorg-x11-libXv-7.2-61  
/usr/lib64/libXv.so.1  
/usr/lib64/libXv.so.1.0.0
```

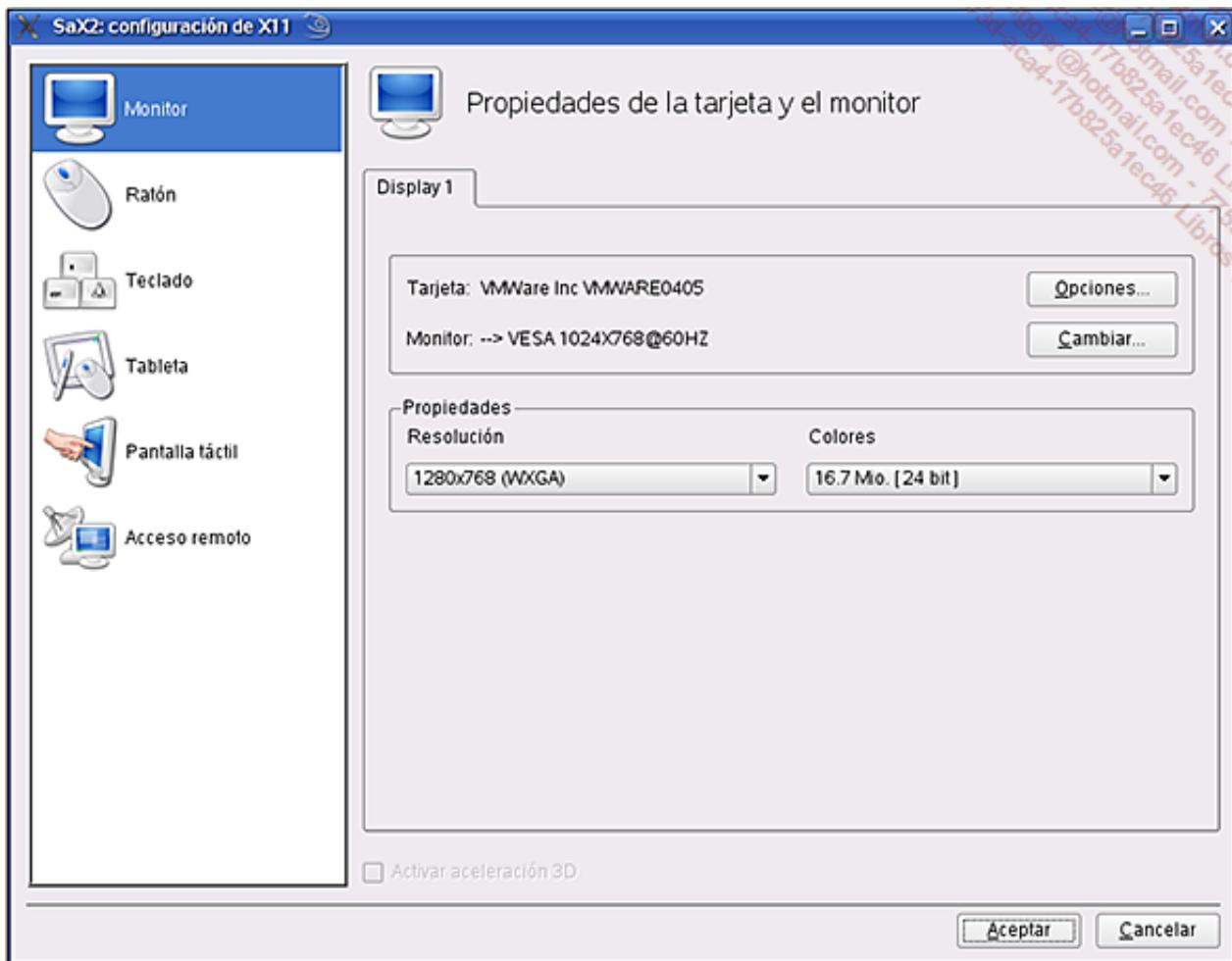
### 3. Configuración

#### a. Por distribución

El fichero de configuración de Xorg está en /etc/X11 y se llama xorg.conf: /etc/X11/xorg.conf. No es habitual, pero sí posible, crear un fichero xorg.conf completamente a mano. En la práctica, la fuerza de las distribuciones Linux reside en parte en su capacidad en detectar su hardware y configurar el entorno gráfico en consecuencia. De hecho, son estas herramientas las que definen, según sus indicaciones y el hardware

detectado, el fichero xorg.conf.

En openSUSE, la herramienta SaX permite configurar el entorno gráfico.



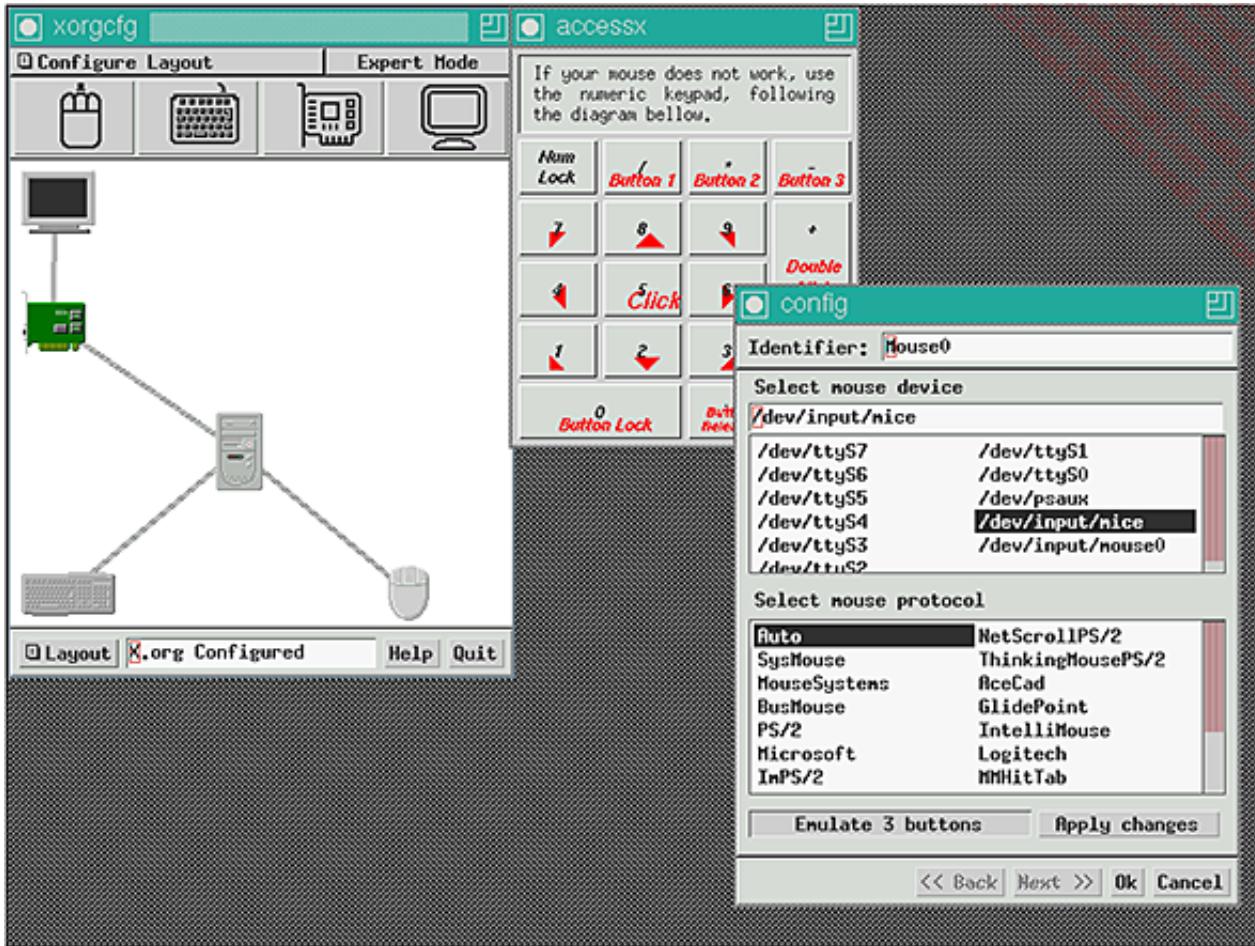
*La herramienta de configuración X SaX2 de openSUSE*

Las últimas versiones de Xorg son autoconfigurables. El servidor X intenta detectar automáticamente su hardware incluso si no hay archivo de configuración. De este modo funciona generalmente bien. Sin embargo, siempre puede modificar usted mismo el archivo de configuración.

## b. Xorgcfg

El comando **xorgcfg** intenta detectar de forma automática todos los parámetros de configuración del servidor X y crear un fichero xorg.conf. En la práctica, una vez que el servidor X dispone de una configuración básica, se inicia X con un gestor de ventanas básico (twm) y una herramienta gráfica que permite modificar, o más bien ajustar, la configuración.

Recuerde que tratamos con una herramienta básica y que luego será preciso modificar la configuración a mano en el fichero resultante. Además, las pruebas realizadas durante la redacción de esta parte del libro han mostrado que xorgcfg es inestable: si usted «mata» Xorg en este momento (con [Alt][Ctrl][Supr]), el ordenador parece reaccionar mal, y quizás haya que reiniciarlo vía el botón reset.



Xorgcfg, herramienta de configuración básica

### c. Xorgconfig

La herramienta **xorgconfig** es similar a xorgcfg, pero está basada en un asistente en modo cuestionario. Equivale y sustituye a la herramienta **xf86config** del antiguo servidor X de XFree86. Tiene que contestar, en orden, a las preguntas siguientes:

- ¿Cuál es el protocolo de comunicación con el ratón?
- ¿Desea una emulación de los tres botones con los otros dos?
- ¿Cuál es el periférico del ratón?
- ¿Cuál es el tipo de teclado?
- ¿En qué idioma?
- ¿Tiene opciones particulares?
- ¿Cuál es su modelo de pantalla (tipo genérico)?
- ¿Cuáles son las frecuencias de barrido de su monitor?
- ¿De cuánta memoria dispone su tarjeta gráfica?
- ¿Cuál es la profundidad de los colores (por ejemplo, 24 bits)?
- ¿Dónde guardar el fichero?

Durante mucho tiempo, este tipo de herramienta era la única disponible para configurar el servidor X. Ahora, las herramientas de las distribuciones lo hacen muy bien. Sin embargo, a pesar de todo debe conocer las bases de la escritura del fichero de configuración `xorg.conf`: las modificaciones manuales pueden ser corrientes en este fichero.

## 4. Estructura de `xorg.conf`

### a. División

El fichero `/etc/X11/xorg.conf` se ordena en forma de secciones y de subsecciones. Cada una corresponde a una funcionalidad del servidor X o a un periférico de entrada o salida.

Para hacer funcionar X, hace falta:

- una salida, una pantalla y la tarjeta gráfica asociada;
- una entrada, teclado, ratón, etc.

En este último caso, la carencia de ratón podría impedir el arranque de todo el sistema. A continuación se detalla el contenido de las diferentes secciones.

### b. Valores booleanos

Determinadas opciones de `xorg.conf` aceptan valores de texto o numéricos, pero muchas sólo toman dos valores. En este caso, son posibles varias opciones:

- Se consideran como VERDADERO los valores 1, on, true o yes.
- Los valores considerados como FALSO serían 0, off, false y no.

### c. Sección `InputDevice`

Una sección **InputDevice** describe un periférico de entrada:

- teclados,
- ratones,
- touchpads,
- pantallas táctiles,
- etc.

Cada periférico de entrada cuenta con su propia sección.

Una sección `InputDevice` se compone de:

- Un identificador único, **Identifier**.
- Un driver, **Driver** (p. ej.: kbd, mouse, evdev, etc.).
- Las diversas opciones, **Option**, relacionadas con el driver.

Las rutas de los ficheros de periféricos se suelen colocar en la sección **Files**, bajo las entradas **InputDevice**.

A continuación le presentamos el ejemplo de la sección que describe un teclado. El driver se llama **kbd**. La sección cuenta con varias opciones que sirven para describir el modelo, el tipo (azerty, qwerty, etc.), el

número de teclas, opciones adicionales, etc. Aquí se describe un teclado español de 105 teclas.

```
Section "InputDevice"
Identifier "Keyboard[0]"
Driver "kbd"
Option "Protocol" "Standard"
Option "XkbLayout" "es"
Option "XkbModel" "pc105"
Option "XkbOptions" "caps:shiftlock"
Option "XkbRules" "xfree86"
EndSection
```

Encontrará ayuda en las manpages, bajo el nombre kbd:

- En Linux el protocolo es siempre **Standard**.
- **XkbModel** define el modelo de teclado. Los teclados recientes toman como valor **pc105**, incluso los teclados con teclas multimedia.
- **XkbLayout** define el idioma del teclado, **es** para un teclado español, **us** para uno estadounidense, **be** para un belga, etc.
- **XkbOptions** facilita opciones específicas. Por ejemplo, **caps:shiftlock** permite obtener el mismo funcionamiento que un teclado en Windows: la tecla CapsLock en Linux permite acceder en principio a las mayúsculas acentuadas, con esta opción accede a las cifras.

Veamos ahora una sección que corresponde al ratón (un modelo de Logitech):

```
Section "InputDevice"
Identifier "Mouse[1]"
Driver "evdev"
Option "Buttons" "10"
Option "InputFashion" "Mouse"
Option "Name" "Logitech USB R*"
Option "Protocol" "event"
Option "SendCoreEvents" "on"
Option "Vendor" "Sysp"
Option "ZAxisMapping" "4 5"
EndSection
```

Podrá observar que la estructura sintáctica es idéntica; sólo cambian el driver y las opciones. Si bien X reconoce la mayoría de los ratones, a menudo hace falta modificar las opciones **Buttons** y **ZaxisMapping**. Además, la interpretación de los eventos de los botones depende de las aplicaciones, entornos de escritorios, etc., y no de X Window.

El driver se llama aquí **evdev**. Un driver genérico es el encargado de gestionar todo tipo de eventos en entrada, tanto para los teclados como los ratones, y sobre todo para estos últimos, ya que es más complicado instalarlo para los teclados. Pero existen drivers específicos: **kbd** es uno de ellos; **mouse**, otro. En cuanto a los dispositivos portátiles, se utiliza a menudo el driver **synaptics** para los touchpads, e incluso el driver **wacom** para los tabletPC y las tabletas gráficas del mismo nombre. A continuación, se muestra una entrada encontrada en un eeePC de Asus:

```
Section "InputDevice"
Identifier "Touchpad"
Driver "synaptics"
Option "SHMConfig" "on"
EndSection
```

## d. Sección Monitor

La sección **Monitor** describe la pantalla de su ordenador. Hoy en día todos los monitores son de tipo DDC, es decir, son capaces de informar mediante el driver gráfico, y por lo tanto al servidor X, las frecuencias y las resoluciones que soportan. Por ello la sección **Monitor** se ve reducida a la mínima expresión.

Sin embargo, si el modelo no está soportado (lo que puede ocurrir teniendo en cuenta el driver gráfico) o si el monitor devuelve información poco fiable, usted puede elegir un modelo genérico, o definir por sí mismo las opciones de su monitor. Para ello, utilice la documentación de éste y busque en particular:

- El tamaño real de la visualización, anchura y altura en milímetros.
- El rango de frecuencias horizontales.
- El rango de frecuencias verticales.
- Si es preciso la lista de las resoluciones soportadas.

La sección del siguiente ejemplo describe un monitor genérico de tipo Vesa que soporta una resolución de 1280x1024 a 60 Hz. Funciona perfectamente con la pantalla LCD de 19 pulgadas 4/3 del autor.

```
Si
Section "Monitor"
    Identifier      "Monitor[0]"
    VendorName     "-> VESA"
    ModelName      "1280X1024@60HZ"
    UseModes        "Modes[0]"
    DisplaySize    340 270
    HorizSync       31.0 - 64.0
    VertRefresh     50.0 - 60.0
    Option          "DPMS"
EndSection
```

su monitor informara de sus frecuencias y de las resoluciones soportadas, y el driver asociado supiera interpretarlas, no necesitaría toda esta descripción. La siguiente sección funciona también muy bien, ya que el driver y el monitor están en la misma onda.

```
Section "Monitor"
    Identifier      "L1915S"
    ModelName      "L1915S"
    VendorName     "LG"
EndSection
```

La mayoría de las opciones hablan por sí mismas. No obstante, le recomendamos que tome buena nota del siguiente punto: UseModes.

## e. Sección Modes

La sección **Modes** o **UseModes** está asociada comúnmente a la sección Monitor. Algunos monitores, y no siempre antiguos o sin marca, necesitan ajustes específicos de frecuencias y de timings para acceder a determinadas resoluciones de visualización. En Windows, estos ajustes son totalmente transparentes para el usuario: o bien el driver es genérico (como los ajustes del primer monitor del punto anterior), o bien la configuración se encuentra en el fichero de descripción «.inf» facilitado con el monitor.

---

 Los monitores recientes (desde hace unos años) no suelen necesitar este tipo de configuración.

---

Xorg y la mayoría de las herramientas de configuración disponen de una base de datos de monitores bastante completa, pero es posible que el suyo no esté presente en ella.

```
Section "Modes"
```

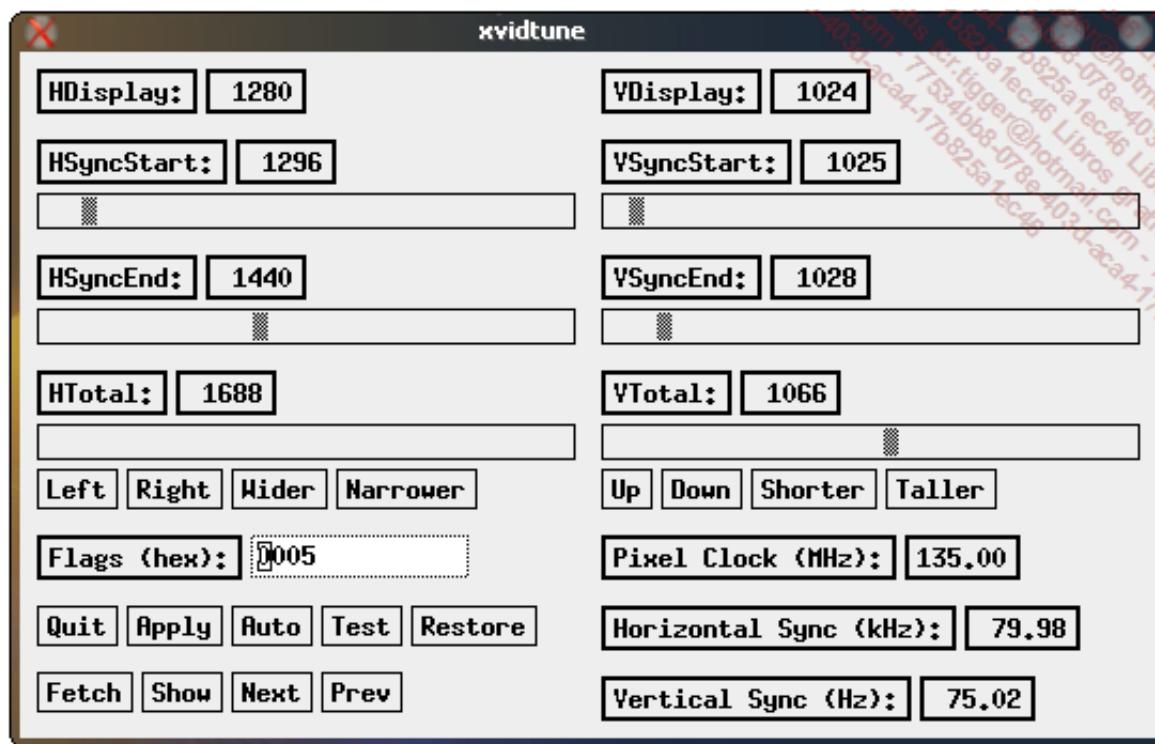
```

Identifier      "Modes[0]"
ModeLine       "1280x1024" 102.6 1280 1312 1472 1632 1024 1028
1032 1048 -hsync -vsync
EndSection

```

No se le pedirá crear *modelines*. Puede encontrar ayuda en las direcciones siguientes:

- <http://www.x.org/wiki/FAQVideoModes> le explica particularmente cómo crear, mediante el registro de eventos de xorg, sus propios *modelines* según la resolución deseada. Aun hace falta que el driver facilite esta información.
- <http://xtiming.sourceforge.net/cgi-bin/xtiming.pl> le permite calcular *modelines* según las especificaciones de su monitor. Puede hacer milagros, pero es arriesgado utilizarlo.
- <http://john.fremlin.de/programs/linux/read-edid/> le proporciona programas que permiten descodificar la información EDID de su monitor.
- [http://gentoo-wiki.com/TIP\\_Getting\\_modelines](http://gentoo-wiki.com/TIP_Getting_modelines) le explica cómo interpretar y utilizar los resultados del comando anterior.



*xvidtune* permite ajustar las resoluciones con mucha exactitud

Si su monitor parece que no funciona con una resolución que se supone que debería aceptar (pantalla negra, de espera, X que sale con el mensaje «no screen found», ninguna resolución aplicable, etc.), piense en comprobar dentro del fichero xorg.conf si la sección Monitor hace llamada a modelines; ásle la línea correspondiente y coméntela. Vuelva a iniciar X ([Alt][Ctrl][Supr]): puede que sea el origen del problema.

Apoyándose en la configuración de X, podría intentar optimizar la visualización de su pantalla. Los casos clásicos (para pantallas CRT, con tubo) son una visualización excéntrica, ni demasiado alta o ancha, o lo contrario. Si los ajustes de su monitor no permiten obtener una buena imagen, puede ayudarse con la herramienta **xvidtune**, que permite influir sobre los ajustes de los modos de vídeo.



¡Cuidado! **xvidtune** puede hacer milagros, pero también puede deteriorar de manera irrevocable su pantalla si usted precisa valores de timings y frecuencias estafalarios. Use esta herramienta con cuidado.

## f. Sección Device

La sección **Device** describe el driver y las opciones de la tarjeta gráfica. Como puede haber varias tarjetas gráficas, puede haber varias secciones de este tipo, con identificadores diferentes. Tenga en cuenta que, con determinados drivers, algunas opciones se colocan en otra sección, **Screen**. El ejemplo siguiente muestra la configuración de una tarjeta gráfica NVidia con el driver propietario.

```
Section "Device"
Identifier "Device[0]"
Driver "nvidia"
VendorName "NVidia"
BoardName "GeForce 8600 GT"
Option "NoLogo" "0"
Option "DPI" "86 x 86"
Option "RenderAccel" "True"
Option "AddARGBGLXVisuals" "True"
EndSection
```

Como para los otros drivers, las opciones dependen del driver y no son siempre las mismas de un driver a otro. Las opciones mostradas arriba son propias del driver Nvidia. La primera permite evitar o no la visualización del logo del fabricante al inicio, y la segunda fuerza el número de DPI de la visualización. Las dos últimas activan la aceleración del display (en el ejemplo, aceleración mixta) y algunas optimizaciones para las extensiones OpenGL.

Aquí tiene un segundo ejemplo con una tarjeta ATI y el driver fglrx propietario:

```
Section "Device"
Identifier "aticonfig-Device[0]"
Driver "fglrx"
Option "XAANoOffscreenPixmaps" "true"
Option "TexturedVideo" "On"
Option "UseFastTLS" "1"
Option "Textured2D" "on"
Option "TexturedXRender" "on"
Option "BackingStore" "on"
Option "VideoOverlay" "Off"
Option "OpenGLOverlay" "Off"
BusID "PCI:1:0:0"
EndSection
```

Para saber más cada una de las opciones, puede remitirse al manual de uso del driver.

## g. Sección Screen

La sección **Screen** es una especie de metasección: llama a las secciones Monitor y Device para agrupar todos los parámetros de visualización. Conociendo las capacidades tanto de la tarjeta gráfica como del monitor, es posible elegir qué resoluciones gráficas deben estar accesibles, para qué número de colores y cuáles son los modos por defecto.

Una sección **Screen** contiene una o varias subsecciones llamadas **Display** que determinan, para un tipo de visualización en n bits (8 bits: 256 colores, 16 bits: 65.536 colores, 24 bits: 16 millones de colores), cuáles deberían ser las resoluciones adecuadas.

- **Depth** define en número de bits la profundidad de los colores.
- **Modes**, las resoluciones soportadas.
- La primera resolución de la lista es la resolución por defecto.
- Puede pasar de una resolución a otra con [Ctrl][Alt] + o - o mediante las posibilidades propuestas por su entorno de escritorio.
- Si no se soporta una resolución, se desactiva automáticamente.

En el ejemplo siguiente, estándar, se han agrupado las secciones anteriores Device y Monitor. La sección Screen se llama Screen[0]. Cuatro subsecciones **Display** configuran la visualización en 8, 15, 16 y 24 bits. Cuatro resoluciones deberían ser accesibles para cada una, la resolución por defecto es de 1.280x1.024.

La entrada **DefaultDepth** precisa la profundidad de los colores por defecto. En este caso, la visualización por defecto será en 1.280x1.024 y 16 millones de colores.

```
Section "Screen"
Identifier      "Screen[0]"
Device         "Device[0]"
Monitor        "Monitor[0]"
DefaultDepth   24
SubSection     "Display"
    Depth       15
    Modes      "1280x1024" "1024x768" "800x600" "640x480"
EndSubSection
SubSection     "Display"
    Depth       16
    Modes      "1280x1024" "1024x768" "800x600" "640x480"
EndSubSection
SubSection     "Display"
    Depth       24
    Modes      "1280x1024" "1024x768" "800x600" "640x480"
EndSubSection
SubSection     "Display"
    Depth       8
    Modes      "1280x1024" "1024x768" "800x600" "640x480"
EndSubSection
EndSection
```

## **h. Sección ServerLayout**

Un fichero xorg.conf dispone de al menos una sección ServerLayout, y en general es la sección básica que agrupa las secciones de visualización y de entrada necesarias para el arranque de X Window y el funcionamiento de la sesión. Se compone de:

- un identificador único, Identifier;
- una o varias entradas **Screen** (en caso de multi-head, doble pantalla o más);
- una o varias entradas **InputDevice** (para el teclado, el ratón, las tabletas, etc.).

X gestiona muy bien las visualizaciones multipantalla y multitarjeta mediante una extensión llamada **Xinerama**. A continuación le presentamos un caso sencillo:

```
Section "ServerLayout"
Identifier      "Layout[all]"
Screen          "Screen[0]" 0 0
InputDevice     "Keyboard[0]" "CoreKeyboard"
InputDevice     "Mouse[1]" "CorePointer"
EndSection
```

---

Detengámonos en lo que sigue al nombre de las secciones **Screen** e **Inputdevice**:

- Los **0 0** después de **Screen** proporcionan la información de posición de la pantalla en caso de modo dual o multi-head. Se trata de las posiciones X e Y partiendo de la esquina superior izquierda.
- El **CoreKeyboard** indica que este periférico de entrada es el teclado principal. Sólo puede haber uno. Si no hay, X busca todo tipo de periféricos de entrada que podría actuar como tal.
- El **CorePointer** es lo mismo, pero para el puntero del ratón.

Si dispone de varios ratones (p. ej.: ratón USB y touchpad) o teclados (p. ej.: un teclado USB enchufado en la base de un portátil, y el teclado del portátil), se pueden declarar los otros periféricos de entradas en la sección **ServerLayout**, pero en este caso escriba «**SendCoreEvents**» y así sucesivamente: esto permite al periférico enviar eventos al servidor X como los otros. X podrá gestionarlos todos:

```
Section "ServerLayout"
Identifier      "Layout[all]"
Screen          "Screen[0]"  0  0
InputDevice     "Keyboard[0]" "CoreKeyboard"
InputDevice     "Mouse[1]"   "CorePointer"
InputDevice     "Touchpad"   "SendCoreEvents"
EndSection
```

## i. Sección Files

La sección **Files** indica al servidor X las rutas hacia algunos ficheros o directorios que pueden serle necesarios:

- los periféricos de entrada,
- las fuentes de caracteres,
- los módulos complementarios,
- la base de los códigos colores RGB (un poco como los códigos HTML).

No hay que establecer algunas rutas, ya que están definidas por defecto. De hecho, y en teoría desde la versión 7.3, no es necesaria ruta alguna, pero puede ser útil precisar otras desconocidas. Las entradas son las siguientes:

- **InputDevices**: la ruta de los periféricos de entrada;
- **FontPath**: la ruta de las fuentes de caracteres;
- **RGBPath**: la ruta del fichero de los códigos RGB;
- **ModulePath**: la ruta de los módulos complementarios de Xorg.

El ejemplo siguiente, procedente de un fichero de configuración Xorg 7.2, precisa las rutas de los periféricos y de las fuentes.

```
Section "Files"
InputDevices    "/dev/gpmdata"
InputDevices    "/dev/input/mice"
FontPath        "/usr/share/fonts/misc:unscaled"
FontPath        "/usr/share/fonts/75dpi:unscaled"
FontPath        "/usr/share/fonts/100dpi:unscaled"
FontPath        "/usr/share/fonts/Type1"
FontPath        "/usr/share/fonts/URW"
FontPath        "/usr/share/fonts/Speedo"
FontPath        "/usr/share/fonts/cyrillic"
```

```

FontPath      "/usr/share/fonts/truetype"
FontPath      "/usr/share/fonts/uni:unscaled"
FontPath      "/opt/kde3/share/fonts"
FontPath      "/usr/local/share/fonts"
EndSection

```

La sección **Files** es optativa. Los fabricantes suelen respetar rutas por defecto y bien conocidas. En particular se buscan las fuentes de caracteres de manera automática en /usr/lib/X11/fonts. No obstante, algunas distribuciones modifican a menudo los scripts de ejecución de X para facilitar otras rutas. En las últimas Mandriva, por ejemplo, las fuentes están todas en /usr/share/fonts.

Las versiones recientes de Xorg van a buscar de manera automática las rutas en el directorio /etc/X11/fontpath.d. Las rutas son vínculos simbólicos hacia los directorios correspondientes de /usr/share/fonts y retoman el formato de las entradas **FontPath**.

## j. Sección Modules

La sección Modules facilita al servidor X una lista de módulos complementarios y optativos que se han de cargar para añadirle nuevas funcionalidades. Se declara un módulo con una línea **Load**.

```

Section "Module"
Load      "dbe"
Load      "type1"
Load      "freetype"
Load      "extmod"
Load      "glx"
EndSection

```

continuación presentamos una lista de módulos habituales. La mayoría son optativos, pero algunos componentes (periféricos de entrada, aplicaciones, tarjetas gráficas) pueden necesitar ciertas extensiones para funcionar. ¿Cómo hacer funcionar OpenOffice.org sin el soporte de las fuentes de caracteres por ejemplo?

- **dbe** (*Double Buffer Extension*) dos búferes de visualización. Uno, principal, al que le corresponde la visualización actual, y otro, secundario, la visualización en segundo plano. Una vez que está listo, se comutan los dos búferes. Este mecanismo evita que la imagen se cuelgue.
- **extmod**: módulo de extensión del protocolo X, que (casi) todo el mundo utiliza. Al final, se debería incorporar en el mismo interior de X.
- **freetype**: permite utilizar fuentes de caracteres TrueType (ttf).
- **type1**: permite utilizar las fuentes Type1.
- **bitmap**: permite utilizar las fuentes bitmap (innecesario para las últimas versiones).
- **GLcore**: modo básico de añadido de las extensiones OpenGL.
- **glx**: extensiones GLX (extensión a OpenGL).
- **dri** (*Direct Rendering Infrastructure*): OpenGL llama a las funciones de la tarjeta gráfica; se acelera considerablemente la visualización en 3D.
- **i2c**: instalación del bus serie i2c, para comunicarse, entre otros, con el monitor.
- **ddc**: protocolo DDC para los monitores (*Display Data Channel*), que pasa por el bus i2c.
- **int10**: capa de emulación/acceso en tiempo real a la interrupción 10 de la tarjeta gráfica, en particular, para acceder a las funcionalidades VESA de la tarjeta y del monitor

- **vbe** (*Vesa Bios Extension*): extensiones Vesa para los accesos a algunos modos y resolución de la tarjeta.

## k. Sección ServerFlags

La sección **ServerFlags** permite definir las opciones globales del servidor X Window. Las opciones son múltiples y aquí le presentamos algunas:

- **DontZap**: si es verdadero, desactiva la secuencia Alt-Ctrl-Backspace (que detiene al servidor X).
- **DontVTSwitch**: si es verdadero, desactiva el acceso a las consolas mediante Alt-Ctrl-Fn.
- **DontZoom**: si es verdadero, prohíbe cambiar la resolución con Alt-Ctrl-+/-.
- **AllowMouseOpenFail**: si es verdadero, X se inicia incluso sin ratón.
- **XkbDisable**: si es verdadero, desactiva el teclado.
- **Xinerama**: activa el soporte dual/multi-head.
- **AIGLX**: activa el soporte AIGLX (depende del driver).
- **BlankTime**: duración en minutos por defecto de la actividad del ahorro de pantalla antes de pasar al modo Stand by de ahorro de energía.
- **StandbyTime**: en minutos, duración de la fase Stand by del ahorro de energía DPMS.
- **SuspendTime**: ídem, pero para la suspensión.
- **Offtime**: ídem, pero para el apagado.

El ejemplo siguiente muestra una sección que permite iniciar X sin ratón (nada impide utilizar luego un ratón USB o inalámbrico), con prohibición de parar el servidor, cambiar de resolución y pasar a consola. Es la configuración por defecto de un nodo de Internet en Linux.

```
Section "ServerFlags"
    Option    "AllowMouseOpenFail" "on"
    Option    "DontZap" "on"
    Option    "DontZoom" "on"
    Option    "DontVTSwitch" "on"
EndSection
```

## I. Sección Extensions

Sección optativa, Extensions permite activar o desactivar extensiones de Xorg. Las más conocidas son Damage y Composite: van juntas.

La extensión Damage permite señalar a las ventanas que se debe volver a dibujar una parte de su visualización. Si se activan las extensiones compuestas (efecto de transparencia, por ejemplo) se puede volver a dibujar una zona de la pantalla mientras permanece cubierta por una ventana cuyo contenido no ha cambiado pero que puede ser transparente: es la de debajo que ha cambiado. La extensión Damage señala el cambio. Suele ser activada por defecto.

La opción compuesta es la que permite tener los efectos del mismo nombre: sombras, transparencia de las ventanas, alpha-blending, etc. Asociada a las 3D, en particular AIGLX, le permite obtener, con los efectos propuestos por Compiz-Fusion, cubos 3D, efectos de escritorio. No siempre es necesario tener una máquina y tarjeta potentes: algunas tarjetas antiguas funcionan muy bien.

```
Section "Extensions"
    Option    "Composite" "Enable"
```

```
| EndSection
```

## m. xorg.conf.d

Hay un método mejor que poner toda la configuración en un único archivo xorg.conf. La configuración se puede modularizar en varios archivos de configuración: uno para los módulos, otro para las entradas y salidas y así en adelante. Estos archivos siguen la misma sintaxis vista hasta ahora. Pueden tener cualquier nombre, pero deben tener ".conf" como sufijo y estar ubicados en /etc/X11/xorg.conf.d o /usr/lib/X11/xorg.conf.d. El manual de xorg.conf proporciona una lista completa de las rutas posibles para estos archivos.

## 5. xorg.conf al completo

A continuación, una configuración completa del fichero xorg.conf:

- Tarjeta Nvidia.
- Pantalla LG 19 pulgadas.
- Resolución de 1.280x1.024.
- 16 millones de colores.
- Teclado español de 105 teclas.
- Ratón con 10 botones.
- Extensiones compuestas activadas.

```
Section "ServerLayout"
Identifier      "Layout[all]"
Screen         "Screen[0]" 0 0
InputDevice    "Keyboard[0]" "CoreKeyboard"
InputDevice    "Mouse[1]" "CorePointer"
EndSection

Section "Files"
InputDevices   "/dev/gpmdata"
InputDevices   "/dev/input/mice"
FontPath       "/usr/share/fonts/misc:unscaled"
FontPath       "/usr/share/fonts/75dpi:unscaled"
FontPath       "/usr/share/fonts/100dpi:unscaled"
FontPath       "/usr/share/fonts/Type1"
FontPath       "/usr/share/fonts/URW"
FontPath       "/usr/share/fonts/Speedo"
FontPath       "/usr/share/fonts/cyrillic"
FontPath       "/usr/share/fonts/truetype"
FontPath       "/usr/share/fonts/uni:unscaled"
FontPath       "/opt/kde3/share/fonts"
FontPath       "/usr/local/share/fonts"
EndSection

Section "Module"
Load          "dbe"
Load          "type1"
Load          "freetype"
Load          "extmod"
Load          "glx"
EndSection

Section "ServerFlags"
```

```
Option      "AllowMouseOpenFail" "on"
Option      "DontZap" "on"
Option      "DontZoom" "on"
Option      "DontVTSwitch" "on"
EndSection

Section "InputDevice"
Identifier    "Keyboard[0]"
Driver        "kbd"
Option        "Protocol" "Standard"
Option        "XkbLayout" "es"
Option        "XkbModel" "pc105"
Option        "XkbOptions" "caps:shiftlock"
Option        "XkbRules" "xfree86"
EndSection

Section "InputDevice"
Identifier    "Mouse[1]"
Driver        "evdev"
Option        "Buttons" "10"
Option        "InputFashion" "Mouse"
Option        "Name" "Logitech USB R*"
Option        "Protocol" "event"
Option        "SendCoreEvents" "on"
Option        "Vendor" "Sysp"
Option        "ZAxisMapping" "4 5"
EndSection

Section "Modes"
Identifier    "Modes[0]"
ModeLine     "1280x1024" 102.6 1280 1312 1472 1632 1024 1028
1032 1048 -hsync -vsync
EndSection

Section "Monitor"
Identifier    "Monitor[0]"
VendorName   "--> VESA"
ModelName    "1280X1024@60HZ"
UseModes     "Modes[0]"
DisplaySize  340 270
HorizSync    31.0 - 64.0
VertRefresh  50.0 - 60.0
Option       "DPMS"
EndSection

Section "Device"
Identifier    "Device[0]"
Driver        "nvidia"
VendorName   "NVidia"
BoardName    "GeForce 8600 GT"
Option        "RenderAccel" "True"
Option        "AddARGBGLXVisuals" "True"
EndSection

Section "Screen"
Identifier    "Screen[0]"
Device        "Device[0]"
Monitor      "Monitor[0]"
DefaultDepth  24
SubSection    "Display"
Depth         15
Modes         "1280x1024"
EndSubSection
SubSection    "Display"
Depth         16
```

```

    Modes      "1280x1024"
EndSubSection
SubSection  "Display"
    Depth     24
    Modes     "1280x1024"
EndSubSection
SubSection  "Display"
    Depth     8
    Modes     "1280x1024"
EndSubSection
EndSection

Section "Extensions"
    Option    "Composite" "Enable"
EndSection

```

## 6. Probar e iniciar X

### a. Comprobar la configuración

#### Xorg 7.2 o anterior

Una vez terminada su configuración de X, es el momento de probar el servidor e iniciararlo. Si tiene una versión de Xorg 7.2 o anterior, siga este proceso. Para ello, obviamente, usted no se debe encontrar ya en modo gráfico. Pase a modo texto (init 2 o 3 según su distribución) y teclee:

```

$ X -probeonly

X Window System Version 7.2.0
Release Date : TRue Jan 22 17:08:26 UTC 2008-05-31
X Protocol Version 11, Revision 0, Release 7.2
Build Operating System: openSUSE SUSE LINUX
Current Operating System : Linux opensuse 2.6.22.17-0.1-default #1
SMP 2008/02/10 20:01:04 UTC i686
Build Date: 22 January 2008
    Before reporting problems, check http://wiki.x.org
    To make sure that you have the lastest version.
Module Loader present
Markers: (--) probed, (**) from config file, (==) default setting,
        (++) from command line, (!!) notice, (II) informational,
        (WW) warning, (EE) error, (NI) not implemented, (??) unknown.
(==) Log file: "/var/log/Xorg.0.log", Time: Sat May 31 10:16:06 2008
(==) Using config file: "/etc/X11/xorg.conf"
(II) Module already build-in

```

No se ha detectado ningún error. De haber habido uno, le saldría esto:

```

(EE) No drivers available.

Fatal server error:
no screens found

```

En este caso:

- O el fichero `/etc/X11/xorg.conf` contiene un error de sintaxis.
- O se ha configurado mal un periférico: pantalla ausente, ninguna resolución soportada, driver incorrecto, etc.

El análisis del registro de sucesos del servidor X es de gran ayuda.

## **Xorg 7.3 o posterior**

Si utiliza una versión 7.3 o posterior de Xorg, puede intentar realizar la autoconfiguración iniciando el comando siguiente:

```
# Xorg --configure
```

Este comando producirá un archivo xorg.conf.new en el directorio de root. Puede modificarlo si así lo desea. Pruebe a continuación este archivo indicando específicamente su uso:

```
# Xorg --config xorg.conf.new
```

Si el servidor X se inicia (puede finalizarlo con [Ctrl][Alt][BackSpace]), renómbrelo y copie este archivo a/**/etc/X11/xorg.conf**. Ésta será la configuración por defecto.

## **b. El registro**

Las entradas en el registro de eventos (trazas) se encuentran en el fichero **/var/log/Xorg.0.log**. Son, a menudo, muy largas: con la configuración de más arriba, el registro contiene 600 trazas y se pueden añadir otras durante el tiempo de funcionamiento. Las entradas en el registro contienen todos los detalles de la carga y de la configuración de X Window.

En caso de error, las últimas líneas suelen ser elocuentes e indican dónde se encuentra el problema. Si X funciona pero no reacciona como debería, busque toda la información en las trazas. Ejemplos de trazas interesantes los proporcionan la tarjeta gráfica y el teclado. Puede ver en negrita qué ha detectado Xorg: una tarjeta Nvidia 8600GT, su versión de BIOS, el bus PCI express, la cantidad de memoria, y la marca y el modelo del monitor.

```
[ 19.835]
X.Org X Server 1.10.1
Release Date: 2011-04-15
[ 19.835] X Protocol Version 11, Revision 0
[ 19.835] Build Operating System: Linux 2.6.24-29-server x86_64 Ubuntu
[ 19.835] Current Operating System: Linux slyserver 2.6.38-11-generic #48-Ubuntu
SMP Fri Jul 29 19:02:55 UTC 2011 x86_64
[ 19.835] Kernel command line: BOOT_IMAGE=/vmlinuz-2.6.38-11-generic
root=UUID=ecb82159-84eb-494e-aalb-41c8f9e9df63 ro quiet splash nomodeset
video=vesafb:mode_option=1280x1024-24,mtrr=3,scroll=ywrap security=selinux selinux=1
[ 19.835] Build Date: 21 May 2011 11:48:41AM
[ 19.835] xorg-server 2:1.10.1-1ubuntu1.1
(For technical support please see http://www.ubuntu.com/support)
[ 19.835] Current version of pixman: 0.20.2
[ 19.835] Before reporting problems, check http://wiki.x.org
      to make sure that you have the latest version.
[ 19.835] Markers: (--) probed, (**) from config file, (==) default setting,
      (++) from command line, (!!) notice, (II) informational,
      (WW) warning, (EE) error, (NI) not implemented, (??) unknown.
[ 19.835] (==) Log file: "/var/log/Xorg.0.log", Time: Tue Aug 23 18:01:40 2011
[ 19.836] (==) Using config file: "/etc/X11/xorg.conf"
[ 19.836] (==) Using system config directory "/usr/share/X11/xorg.conf.d"
[ 19.836] (==) No Layout section. Using the first Screen section.
[ 19.836] (**) |-->Screen "Default Screen" (0)
...
[ 19.900] (**) NVIDIA(0): Depth 24, (--) framebuffer bpp 32
[ 19.900] (==) NVIDIA(0): RGB weight 888
[ 19.900] (==) NVIDIA(0): Default visual is TrueColor
[ 19.900] (==) NVIDIA(0): Using gamma correction (1.0, 1.0, 1.0)
[ 19.900] (**) NVIDIA(0): Option "NoLogo" "True"
[ 20.687] (II) NVIDIA(GPU-0): Display (LG Electronics L1915S (CRT-0)) does not support
[ 20.687] (II) NVIDIA(GPU-0):      NVIDIA 3D Vision stereo.
```

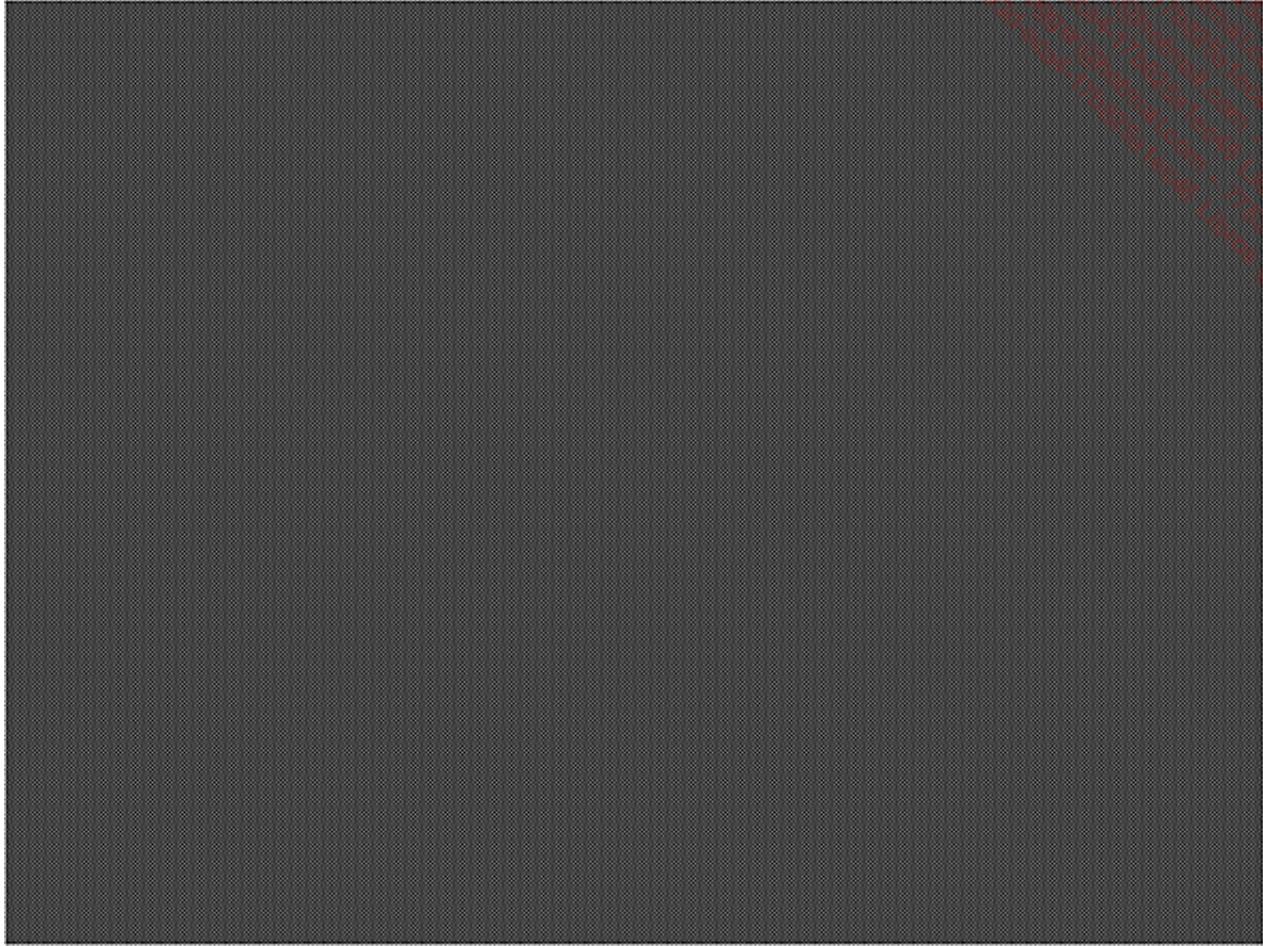
```
[ 20.690] (II) NVIDIA(0): NVIDIA GPU GeForce 8600 GT (G84) at PCI:1:0:0 (GPU-0)
[ 20.690] (--) NVIDIA(0): Memory: 262144 kBytes
[ 20.690] (--) NVIDIA(0): VideoBIOS: 60.84.35.00.11
[ 20.690] (II) NVIDIA(0): Detected PCI Express Link width: 16X
[ 20.690] (--) NVIDIA(0): Interlaced video modes are supported on this GPU
[ 20.690] (--) NVIDIA(0): Connected display device(s) on GeForce 8600 GT at PCI:1:0:0
...
[ 21.051] (II) Using input driver 'evdev' for 'USB Video Camera'
[ 21.051] (II) Loading /usr/lib/xorg/modules/input/evdev_drv.so
[ 21.051] (**) USB Video Camera: always reports core events
[ 21.051] (**) USB Video Camera: Device: "/dev/input/event5"
[ 21.090] (--) USB Video Camera: Found keys
[ 21.090] (II) USB Video Camera: Configuring as keyboard
[ 21.090] (**) Option "config_info"
"udev:/sys/devices/pci0000:00/0000:00:1a.7/usb1/1-5/1-5:1.0/input/input5/event5"
[ 21.090] (II) XINPUT: Adding extended input device "USB Video Camera" (type: KEYBOARD)
[ 21.090] (**) Option "xkb_rules" "evdev"
[ 21.090] (**) Option "xkb_model" "pc105"
[ 21.090] (**) Option "xkb_layout" "fr"
[ 21.090] (**) Option "xkb_variant" "oss"
[ 21.091] (II) config/udev: Adding input device HDA Intel Headphone
(/dev/input/event6)
...
```

### c. Probar el servidor

Prueba el servidor X iniciándolo con su nombre:

```
$ X
```

Si X funciona, debe tener una pantalla gris, en realidad una sucesión de puntos negros y blancos, y una cruz en medio que, es el cursor del ratón. Mueva el ratón. Si funciona, es que se ha configurado bien su servidor.



*Una pantalla gris: X funciona*

Detenga el servidor X con [Ctrl][Alt][Retroceso], porque sin ningún gestor de ventanas no hay nada que hacer.

# El Display Manager

## 1. Principio

El Display Manager, o gestor de visualización, es un elemento de X Window que se encarga de la conexión de usuarios, locales o remotos, de su autenticación, y de la carga de componentes en su entorno de trabajo al abrir sesión. Gestiona un conjunto de visualizaciones X que pueden ser locales o remotas.

Para las conexiones remotas, el Display Manager se apoya en el protocolo estándar XDMCP: X Display Manager Control Protocol.

El Display Manager representa más o menos el equivalente gráfico de los servicios propuestos por init, getty y login: pide identificadores y contraseñas, autentica a las personas y abre una sesión.

El gestor por defecto se llama XDM: X Display Manager. Su estilo gráfico no resulta muy atractivo, pero es ligero y funciona con todos los servidores X.

Se puede ver una sesión X como una sesión de consola: es la duración del proceso de un usuario después de la conexión. En una consola es el shell; en X, es una «session manager», en general, el gestor de ventanas (o un proceso del entorno ofimático) o un terminal gráfico. Cuando este proceso está cerrado (desconexión del terminal, salida del entorno ofimático, etc.) la sesión finaliza y el sistema vuelve a mostrar la caja de conexión de XDM (o cualquier otro Display Manager).

Es posible abrir varias sesiones X desde un solo Display Manager, incluso en una misma máquina.

Linux dispone de varios Display Manager, pero se utilizan principalmente tres:

- XDM: versión básica.
- GDM: versión propuesta por GNOME.
- KDM: versión propuesta por KDE.

Las versiones GNOME y KDE han evolucionado: proponen lo mismo que XDM con funcionalidades adicionales:

- lista de los usuarios;
- iconos (avatares) asociados;
- elección de una sesión gráfica particular (Gnome, fvwm, KDE, etc.);
- posibilidad de autoconexión;
- temas gráficos atractivos;
- lista de los servidores X remotos (XDMCP);
- pasar de un usuario a otro;
- etc.

## 2. XDM

### a. Configuración general

Los ficheros de configuración de XDM se encuentran en el directorio `/etc/X11/xdm`. El primer fichero cargado es `xmdm-config`. Todas sus líneas son del tipo:

**DisplayManager.recurso:** valor.

Cada línea representa un recurso de XDM. La certificación LPI1 no exige saber de memoria la configuración de XDM, sólo algunos elementos básicos, como xdm-config, que carga otros ficheros. En el ejemplo siguiente, las líneas en negrita muestran algunos otros scripts de configuración llamados, o interpretados directamente por xdm, o ejecutados antes o después de la conexión:

```
!! xdm-config: Configuration of the xdm
!
DisplayManager.errorLogFile:          /var/log/xdm.errors
DisplayManager.pidFile:              /var/run/xdm.pid
DisplayManager.authDir:              /var/lib/xdm
DisplayManager.keyFile:              /etc/X11/xdm/xdm-keys
DisplayManager.servers:           /etc/X11/xdm/Xservers
DisplayManager.accessFile:         /etc/X11/xdm/Xaccess
DisplayManager.willing:             su nobody -c /etc/X11/xdm/Xwilling
!
! ATTENTION: `authName' should be in general MIT-MAGIC-COOKIE-1
! For XDM-AUTHENTICATION-1 which is default for xterminals see
! manual page of xdm and the manual coming with the xterminal.
!
DisplayManager.*.authName:          MIT-MAGIC-COOKIE-1
DisplayManager.*.authComplain:      false
!
! All displays should use authorization, but we cannot be sure
! X terminals will be configured that way, so by default
! use authorization only for local displays :0, :1, etc.
!
DisplayManager._0.authorize:        true
DisplayManager._1.authorize:        true
DisplayManager._93.authorize:       true
!
! The scripts handling the setup, the startup, the session its self,
! and the reset of an X session.
!
DisplayManager.*.setup:          /etc/X11/xdm/Xsetup
DisplayManager.*.chooser:         /etc/X11/xdm/RunChooser
DisplayManager.*.startup:         /etc/X11/xdm/Xstartup
DisplayManager.*.session:         /etc/X11/xdm/Xsession
DisplayManager.*.reset:          /etc/X11/xdm/Xreset
!
DisplayManager._0.terminateServer:   true
DisplayManager._93.terminateServer:  true
!
DisplayManager*resources:         /etc/X11/xdm/Xresources
DisplayManager.*.terminateServer:    false
!
! SECURITY: do not listen for XDMCP or Chooser requests
! Comment out this line if you want to manage X terminals with xdm
!
!DisplayManager.requestPort:        0
```

## b. Setup: Xsetup

El fichero **/etc/X11/xdm/Xsetup** o cualquier otro asociado a la entrada **DisplayManager.\*.setup** es invocado antes incluso de que el sistema muestre la ventana de login. Es en este fichero donde puede determinar la presentación gráfica de XDM o cómo visualizar nuevas ventanas (registro del sistema, relojes, etc.) El contenido por defecto de este fichero depende del editor de la distribución, pero en general todos ellos lo usan para:

- modificar el fondo de pantalla con el comando xsetroot;

- modificar los colores, con xrdb;
- cambiar con los recursos X la geometría (las posiciones y tamaños) de los cuadros de diálogo;
- visualizar una consola con xconsole;
- activar el teclado numérico con numlockx;
- modificar el teclado con xmodmap;
- etc.

Aquí tiene un ejemplo depurado procedente una vez más de openSUSE. Las líneas en negrita muestran las acciones para xdm.

```

#
# Handle background:
# First kdm/gdm choose, then xdm/user choose and
# if no choose is given use the system defaults.
# Pantalla de selección
if test "$kdm" = "yes" -o "$gdm" = "yes" ; then
    : # $xsetroot -solid '#738dc6'
elif test -s ${background}.gz -a -x $xpmroot ; then
    $xpmroot ${background}.gz
elif test -s ${background} -a -x $xpmroot ; then
    $xpmroot $background
elif test -x $backprg ; then
    $backprg
else
    $xsetroot -gray
fi

#
# Enable Numlock if set
# Activación de la extensión del teclado numérico
if test -r /var/run/numlock-on && type -p numlockx > /dev/null ;
then
    numlockx on
fi

#
# Xresources
# Modificación de colores y geometría de la pantalla
if test "$kdm" != "yes" -a "$gdm" != "yes" ; then
    $xrdb -override -retain <<-EOF
    #ifdef COLOR
    *Foreground: black
    *Background: #cdd2b4
    #endif
    #if (WIDTH < 320)
    XConsole*geometry: 125x80+0-0
    #elif (WIDTH < 400)
    XConsole*geometry: 130x85+0-0
    #elif (WIDTH < 640)
    XConsole*geometry: 150x90+0-0
    #elif (WIDTH < 800)
    XConsole*geometry: 240x95+0-0
    #elif (WIDTH < 1024)
    XConsole*geometry: 300x100+0-0
    #elif (WIDTH < 1152)
    XConsole*geometry: 384x110+0-0
    #elif (WIDTH < 1280)
    XConsole*geometry: 432x120+0-0
    #else

```

```

XConsole*geometry: 480x130+0+0
#endif
EOF
fi
if test "$kdm" = "yes" -o "$gdm" = "yes" ; then
    $xrdb -override -retain ${ETCDIR}/Xresourcesf
i
#
# The geometry of xconsole is set in the Xresource file.
# Inicio de una consola
(
    exec setsid $xconsole -notify -nostdin -verbose -exitOnFail
) & echo $! > /var/run/xconsole.pid

```

### c. Chooser: RunChooser

El fichero `/etc/X11/xdm/RunChooser` o cualquier otro asociado con la entrada **DisplayManager.\*.chooser** permite visualizar el cuadro de diálogo de bienvenida de los servidores X remotos para conectarse a ellos. Se parece al fichero Xsetup, pero se limita a la configuración y al inicio del programa **chooser** (`/usr/X11R6/bin/chooser`, `/usr/lib/X11/chooser` o cualquier otro programa que tenga una función similar). Este programa tiene interés únicamente si el servidor X remoto acepta las conexiones mediante XDMCP.

### d. Startup: Xstartup

Tras la autenticación del usuario, el sistema busca el fichero `/etc/X11/xdm/Xstartup` o cualquier otro asociado con la entrada **DisplayManager.\*.startup** y lo ejecuta. Se ejecuta con privilegios de root. No debe confundirse con el fichero de sesión, del cual hablaremos más adelante. Sirve, entre otras cosas, para:

- borrar la pantalla;
- escribir la información de conexión en los ficheros `/var/log` adecuados;
- comprobar si la conexión es local o remota;
- comprobar si se autoriza al usuario a conectarse;
- etc.

### e. Sesión: Xsession

Tras la autenticación del usuario y ya con el conjunto de privilegios de éste, se inicia el fichero `/etc/X11/xdm/Xsession` o cualquier otro asociado con la entrada **DisplayManager.\*.session**. El sistema comprobará la instalación del servidor X; si no está instalado, ejecutará un comando **Xterm** y lo asociará el proceso de sesión del usuario. De hecho, si una sesión de usuario no se pudiera iniciar con normalidad, este fichero contiene lo necesario para iniciar una consola. Después de una configuración básica, Xsession intenta iniciar en este orden los ficheros siguientes:

- el fichero `$HOME/.xsession`, propio del usuario;
- si no lo encuentra, el fichero `$HOME/.xinitrc` del usuario;
- si no lo encuentra, el fichero `/etc/X11/xdm/sys.xsession`;
- si no lo encuentra, el fichero `/etc/X11/xinit/xinitrc`.

Mediante una bandera (flag) se puede forzar un arranque por defecto e ir directamente a la carga de un Window Manager (KDE o GNOME) sin pasar por los ficheros anteriores. Los entornos gráficos más evolucionados tienen automatizada su propia configuración.

```

#
# Forced X session type if the user asked for
# an other session environment.
#
if test "$forced" = "yes" ; then
    unset WINDOW_MANAGER STARTUP
    test -x $syssess && exec_login "$syssess"
    exec_login "/bin/bash $syssess"
fi

# User login X session
# If the user doesn't have their own xsession, then run
# system xsession or xinitrc script if they exist

if test -f $session ; then
    test -x $session && exec_login "$session"
    exec_login "/bin/bash $session"
elif test -f $xinitrc ; then
    test -x $xinitrc && exec_login "$xinitrc"
    exec_login "/bin/bash $xinitrc"
elif test -f $syssess; then
    test -x $syssess && exec_login "$syssess"
    exec_login "/bin/bash $syssess"
elif test -f $sysinit ; then
    test -x $sysinit && exec_login "$sysinit"
    exec_login "/bin/bash $sysinit"
elif test -n "$WINDOWMANAGER" ; then
    unset WINDOW_MANAGER STARTUP
    exec_login "$WINDOWMANAGER"
fi

```

## f. Reset: Xreset

Cuando un usuario cierra la sesión, el sistema inicia el fichero `/etc/X11/xdm/Xreset` o cualquier otro asociado con la entrada **DisplayManager.\*.reset**. En el registro del sistema encontrará entradas procedentes de este fichero.

## g. Resources: Xresources

El fichero `/etc/X11/xdm/Xresources` o cualquier otro asociado con la entrada **DisplayManager\*resources** con xdm contiene las definiciones de los recursos visuales utilizados por xdm, como las fuentes de caracteres, los mensajes de bienvenida o error, los colores, la adaptación de la visualización en función del número de colores y el tamaño de la pantalla, etc.

En particular, en este fichero usted puede modificar el mensaje de bienvenida o de error:

```

xlogin*titleMessage: Xlogin
xlogin*greetColor: darkgray
xlogin*promptColor: darkgray
xlogin*failColor: red
xlogin*greeting: Bienvenido/a a CLIENTHOST
xlogin*fail: -- Conexión denegada -

```

Con una configuración como la anterior, el sistema recibirá al usuario con el mensaje «Bienvenido/a a CLIENTHOST» visualizado en gris oscuro. Si no se autentica, se visualiza en rojo el mensaje «-- Conexión denegada».

## h. Servers: Xservers

El fichero `/etc/X11/xdm/Xservers` o cualquier otro fichero asociado con la entrada **DisplayManager.servers** proporciona la lista de las especificaciones de los servidores locales de X o, más bien, de los que no necesitan conexión mediante XDMCP. La línea siguiente indica que el primer servidor :0 es local, su binario es /usr/bin/X con opción -br vt7 (ocupa el terminal virtual vt7).

```
:0 local /usr/bin/X -br vt7
```

En esta línea, puede, si lo desea, modificar el número de colores del servidor X durante el inicio de xdm pasándole los argumentos correctos. Para cambiar a 256 colores:

```
:0 local /usr/bin/X -br vt7 -depth 8
```

### i. AccessFile: Xaccess y XDMCP

El fichero `/etc/X11/xdm/Xaccess` o cualquier otro fichero asociado con la entrada **DisplayManager.accessFile** proporciona la lista de los anfitriones autorizados a conectarse mediante XDMCP a su servidor X. Para autorizar conexiones, tendrá que modificar dos ficheros:

- En `/etc/X11/xdm/xdm-config`, quite el comentario a la línea '`!DisplayManager.requestPort`', de tal manera que quede como a continuación:

```
$ grep request /etc/X11/xdm/xdm-config
...
DisplayManager.requestPort: 0
```

- En `/etc/X11/xdm/Xaccess`, quite el comentario a la línea que empieza con un asterisco:

```
* CHOOSEN BROADCAST
```

O aún más fácil:

```
*
```

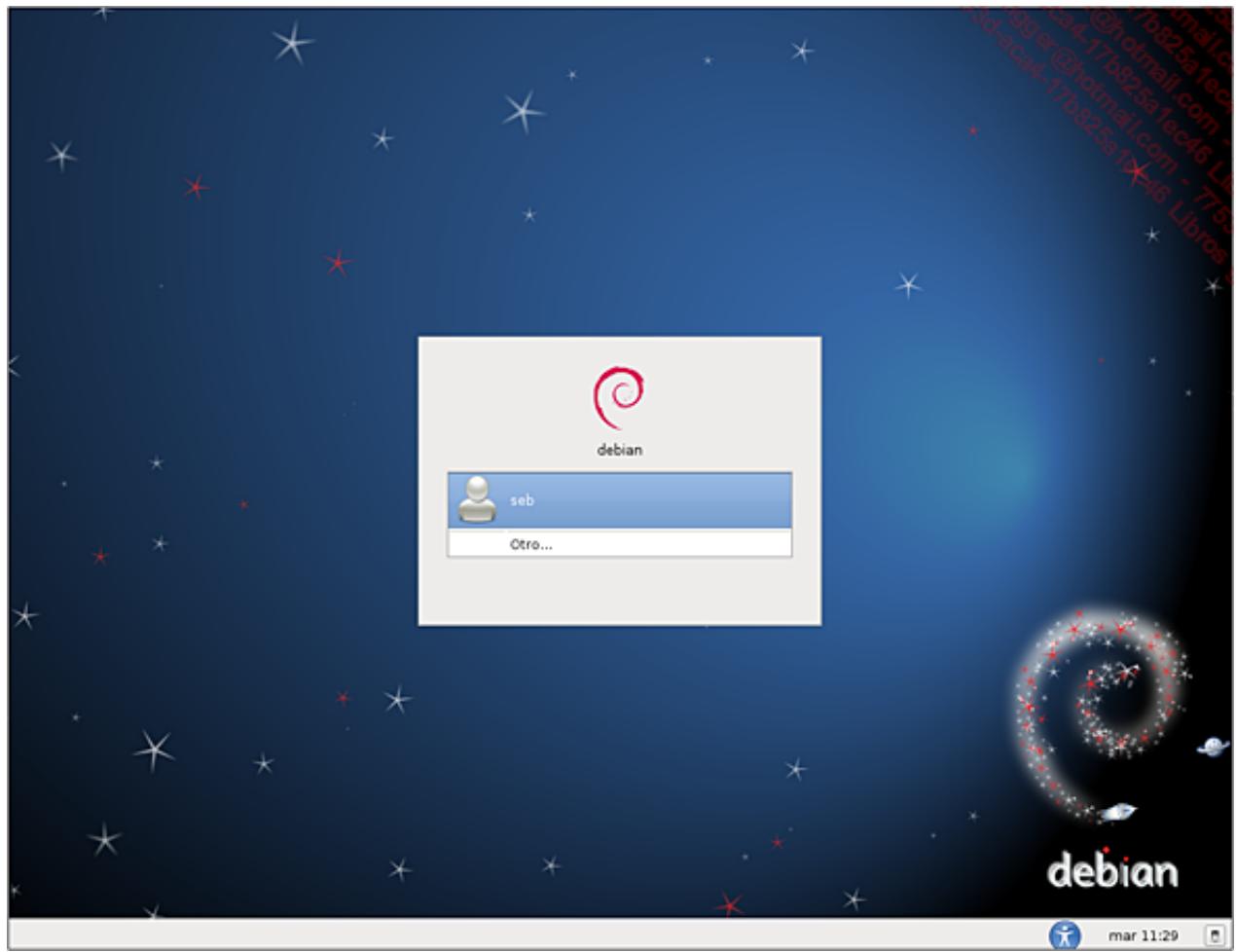
Esto es todo. Todos los anfitriones podrán conectarse a su máquina.

## 3. gdm y kdm

**gdm** y **kdm** son, respectivamente, los gestores de visualización de GNOME y KDE.

¿Por qué hemos dedicado tanto tiempo a xdm cuando gdm y kdm son mucho más bonitos, prácticos y eficaces (aquí puede usted añadir tantos piropos superlativos como quiera...)? Porque, excepto por supuesto para todos los recursos gráficos propios, gdm y kdm emplean (aunque no sea una obligación) los mismos ficheros que xdm, entre los cuales figuran:

- Xsetup
- Xstartup
- Xsession
- Xreset
- Xresources
- Xaccess
- etc.



GDM, el Display Manager de Gnome

La configuración avanzada de gdm y kdm se salta los ficheros comunes de XDM (que configura de forma automática), pero permite configurar manualmente aquéllos propios del gestor, ya sea mediante ficheros de texto o, si las modificaciones son sencillas, pasando por la interfaz gráfica asociada.

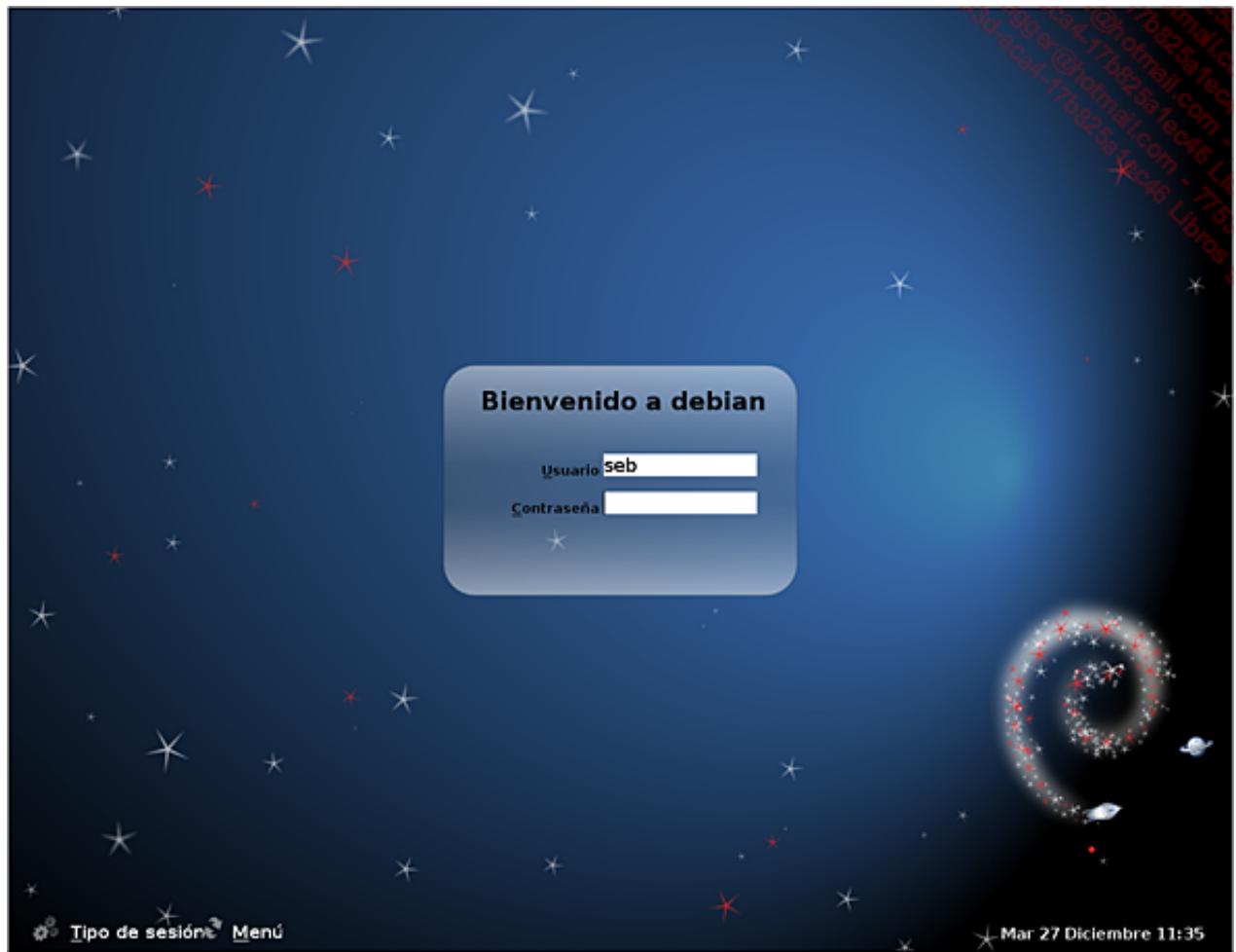
- La configuración de KDM se coloca en `/etc/X11/kdm/kdmrc` o en `<prefix-kde>/share/config/kdmrc`, por ejemplo en openSUSE el fichero se coloca en `/opt/kde3/share/config/kdm/kdmrc`.
- La configuración de GDM se coloca en `/etc/X11/gdm/gdm.conf`, o `/etc/gdm/gdm.conf` o también en `/usr/share/gdm/gdm.conf`.

Aunque gdm y kdm utilizan por defecto las opciones de los ficheros de XDM, puede modificar la configuración de manera que se abstraiga totalmente de ellas. Por ejemplo, para kdm, puede activar XDMCP así:

```
[Xdmcp]
Enable=true
Port=177
KeyFile=/etc/kde4/kdm/kdmkeys
Xaccess=/etc/kde4/kdm/Xaccess
```

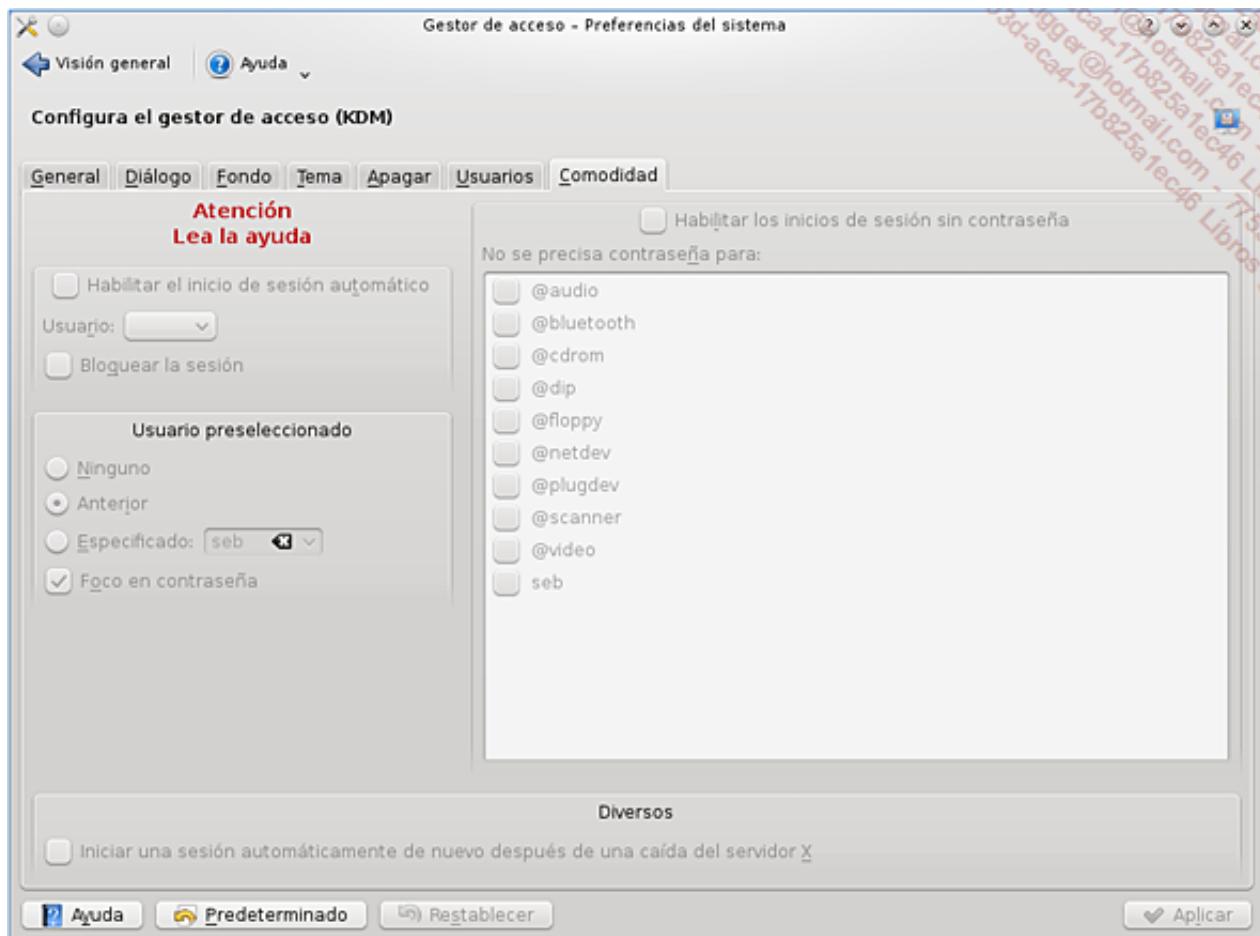
Otra (falsa) buena idea es autorizar la exportación de las ventanas X hacia su servidor X (visualización remota) mediante el comando **xhost+** suprimiendo el parámetro **nolisten** de la configuración. Comente la línea siguiente:

```
ServerArgsLocal=-nolisten tcp
```



*KDM, iniciado desde Debian 6*

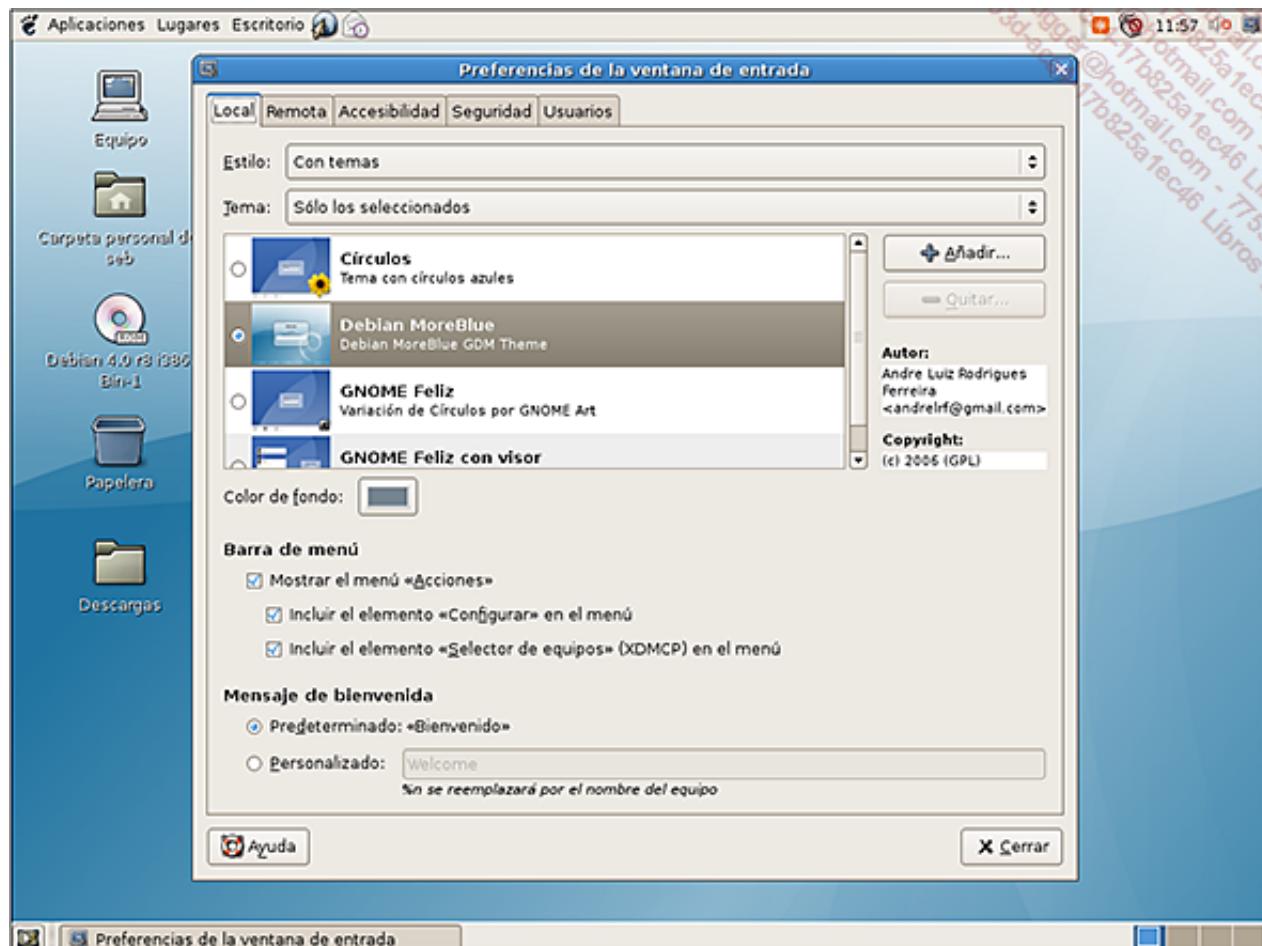
Para el resto, use la herramienta de configuración gráfica. Inicie el **Centro de configuración de KDE**, y despliegue la entrada **Administración del sistema**. Haga clic en **Gestor de conexión**. Como usuario genérico no podrá hacer nada: haga clic en el botón **Modo superusuario**, abajo.



### *Personalización de KDM en KDE4*

A partir de ese punto, puede configurar kdm completamente.

En gdm ejecute, desde la línea de comandos, **gdmsetup**, que inicia la ventana de configuraciones. El ejemplo siguiente procede de una instalación Debian Etch básica.



Personalización de GDM

## 4. xdm, gdm o kdm al boot

### a. inittab

Para iniciar X y el gestor de ventanas durante el proceso de arranque del sistema, los administradores cuentan con dos opciones en la mayoría de las distribuciones:

- editar `/etc/inittab`;
- o iniciar xdm como servicio.

En el primer caso, abra el fichero `/etc/inittab` y busque una línea que se parezca a una de las líneas siguientes. Encontrará tanto las líneas once como respawn, siendo preferible el segundo caso:

```
xdm:5:once:/usr/X11R6/bin/xdm
xdm:5:respawn:/usr/X11R6/bin/xdm
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Si  
el

fichero **inittab** contiene una línea equivalente a la del ejemplo pero usted sigue accediendo al sistema en modo consola, puede ser debido a que:

- X está mal configurado, y por lo tanto después de algunos intentos devuelve el control a la consola. En este caso, la pantalla debería parpadear varias veces en este momento;

- usted está en el nivel de ejecución (runlevel) incorrecto.

En este último caso, compruebe y modifique la línea siguiente para iniciar en nivel 5 (para las distribuciones rpm) por defecto:

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

y pase como root al nivel de ejecución 5:

```
# telinit 5
```

## b. Servicio

Alternativamente se puede iniciar `xdm` como servicio. Verifique la siguiente línea en `/etc/init.d`:

```
$ ls /etc/init.d/xdm
/etc/init.d/xdm
```

Verifique la configuración del nivel asociado mediante **rcupdate.d** o **chkconfig**, o manualmente:

```
# chkconfig --list xdm
xdm      0:off 1:off 2:off 3:off 4:off 5:on 6:off
# ls -l /etc/rc.d/rc?.d/S*xdm
lrwxrwxrwx 1 root root 6 may 14 12:22 /etc/rc.d/rc5.d/S10xdm ->
./xdm
```

## c. /etc/sysconfig

En varias distribuciones, se colocan la elección y los ajustes por defecto de varios gestores de ventanas en ficheros de configuración dentro de la carpeta `/etc/sysconfig`. En Mandriva, por ejemplo, se trata del fichero `/etc/sysconfig/desktop`.

```
$ cat /etc/sysconfig/desktop
DISPLAYMANAGER=kdm
```

Con openSUSE, los ajustes son más precisos (en particular para la gestión de la resolución y de los temas) y el fichero es `/etc/sysconfig/displaymanager`. Particularmente encontrará ajustes que sustituyen a los de xdm relativos a los mensajes que se van a visualizar, los accesos XDMCP, etc.

```
$ cat /etc/sysconfig/displaymanager | grep -v ^#
DISPLAYMANAGER_XSERVER=Xorg
DISPLAYMANAGER_XGL_OPTS="-accel glx:pbuffer -accel xv:pbuffer"
DISPLAYMANAGER="kdm"
DISPLAYMANAGER_REMOTE_ACCESS="no"
DISPLAYMANAGER_ROOT_LOGIN_REMOTE="no"
DISPLAYMANAGER_STARTS_XSERVER="yes"
DISPLAYMANAGER_XSERVER_TCP_PORT_6000_OPEN="no"
DISPLAYMANAGER_AUTOLOGIN=""
DISPLAYMANAGER_PASSWORD_LESS_LOGIN="no"
DISPLAYMANAGER_AD_INTEGRATION="no"
DISPLAYMANAGER_SHUTDOWN="auto"
DISPLAYMANAGER_RANDR_MODE_VGA="auto"
DISPLAYMANAGER_RANDR_MODE_auto="1024x768_60 64.11 1024 1080 1184
1344 768 769 772 795 -HSync +Vsync"
KDM_USERS=""
KDM_GREETSTRING=""
DISPLAYMANAGER_KDM_THEME="SUSE"
```

#### **d. Ubuntu y Debian**

Ubuntu y Debian disponen de servicios gdm (o gdm3) y kdm en `/etc/init.d`, que se activan generalmente por defecto. Para modificar el gestor de sesiones por defecto, edite el archivo `/etc/X11/default-display-manager` e indique en la única línea de este archivo la ruta del ejecutable asociado:

```
# cat /etc/X11/default-display-manager  
/usr/bin/kdm
```

# Window Manager y entorno personal

## 1. Mediante el display manager

Usted suele determinar el tipo de entorno gráfico cargado por la sesión X desde gdm o kdm (no es habitual utilizar xdm) y el menú asociado. Éstos recuerdan la sesión anterior como sesión por defecto gracias a la presencia del fichero `.dmrc` en su directorio personal:

```
$ pwd  
/home/seb  
$ cat .dmrc  
~$ cat .dmrc  
  
[Desktop]  
Session=kde-plasma  
Language=es_ES.utf8  
Layout=es
```

Tanto si utiliza como gestor gdm o como si emplea kdm, notará que ambos comparten la lista de las sesiones posibles. Es una de las ventajas de Freedesktop: los ficheros de descripción de las sesiones están ubicados en `/usr/share/xsessions` (la descripción del fichero kdmrc configura otras ubicaciones alternativas).

```
$ pwd  
/usr/share/xsessions  
$ ls -ltotal 20  
-rw-r--r-- 1 root root 7262 oct 4 2007 fvwm2.desktop  
-rw-r--r-- 1 root root 6796 may 28 13:44 kde.desktop  
-rw-r--r-- 1 root root 3305 nov 28 2007 twm.desktop
```

Cada fichero **desktop** representa una sesión posible. Las descripciones y los comandos de inicio están presente en cada uno de los ficheros, y en particular mediante las líneas Exec y Name:

```
$ grep -E "Exec|Name=" *.desktop  
fvwm2.desktop:Exec=fvwm  
fvwm2.desktop:TryExec=fvwm  
fvwm2.desktop:Name=FVWM  
kde.desktop:Exec=/opt/kde3/bin/startkde  
kde.desktop:TryExec=/opt/kde3/bin/startkde  
kde.desktop:Name=KDE3  
twm.desktop:Exec=twm  
twm.desktop:TryExec=twm  
twm.desktop:Name=TWM
```

Cuando selecciona una sesión particular en gdm o kdm, éste lista las sesiones en `/usr/share/xsessions` y propone la lista. Cuando el usuario se conecta, se escribe su elección como sesión por defecto en su fichero `.dmrc`; luego se ejecuta el comando correspondiente a la línea **Exec**.

## 2. startx

Si la sesión actual se encuentra en modo comando y tiene instalado el servidor X, puede usar el comando **startx** para iniciar una sesión del servidor X y el entorno asociado.

```
$ startx
```

Cuando el sistema no encuentra ningún fichero de configuración, visualizará X, una pantalla gris con un cursor de ratón que no ofrece acción alguna.

Cuando usted inicia X con **startx**, el sistema intenta ejecutar en este orden:

- el fichero **\$HOME/.xinitrc**;
- **/etc/X11/xinit/xinitrc**.

Dado que es recomendable no tocar el fichero **/etc/X11/xinit/xinitrc** para que se mantenga fiel al estándar definido por su distribución, usted puede modificar su fichero personal **.xinitrc**. Como son iguales, puede usar el primero como modelo del segundo. Aquí tiene un ejemplo de **.xinitrc** en Mandriva:

```
$ cat xinitrc
#!/bin/sh

if [ "`whoami`" != root ]; then
    xsetroot -solid "#21449C"
fi

exec /etc/X11/Xsession $*
```

La última línea ejecuta Xsession, que va a buscar un gestor de ventanas para iniciar. Pero un simple fichero que contenga una única línea puede bastar para iniciar KDE, siempre en Mandriva:

```
$ cat $HOME/.xinitrc
/usr/bin/startkde
```

En openSUSE:

```
/opt/kde3/bin/startkde
```

para lanzar GNOME:

```
exec gnome-session
```

### 3. Los terminales

Para acceder a línea de comandos desde X, debe utilizar un emulador de terminal. Son numerosos: ¿cuál elegir?

En primer lugar, tenemos al emulador básico del servidor X llamado **xterm**. Sea cual sea su entorno de trabajo X, siempre estará presente. Por eso, el sistema lo inicia por defecto cuando encuentra problemas con el servidor X.

```
$ xterm &
```

```

X seb@slyserver:~/Documents/slyunix
-rw-r--r-- 1 seb users 158849 mar  2 2004 pub_planete.jpg
-rw-r--r-- 1 seb users 78135 mar  2 2004 pub_planete.pdf
-rw-r--r-- 1 seb users 5636322 mar  2 2004 pub_planete.sxd
-rw-r--r-- 1 seb users 38957 feb 26 2004 pub.sla
-rwxr--r-- 1 seb users 473361 jun  3 2004 PubSlyunix.JPG
-rwxr--r-- 1 seb users 226116 jun  3 2004 PubSlyunix.PDF
-rw-r--r-- 1 seb users 7099 ago  3 2004 session_slyunix.sxw
-rw-r--r-- 1 seb users 155118 dic  5 2004 site_1.jpg
-rw-r--r-- 1 seb users 132489 dic  5 2004 site_2.jpg
-rw-r--r-- 1 seb users 137897 dic  5 2004 site_3.jpg
drwxr-xr-x 26 seb users 4096 sep 13 2007 SITE_MW
-rw-r--r-- 1 seb users 125952 nov 16 2003 Slyunix.doc
-rw-r--r-- 1 seb users 721618 oct 31 2003 Slyunix.pdf
-rw-r--r-- 1 seb users 486072 feb  7 2004 Slyunix.sxw
-rw----- 1 seb users 644090 jun  3 2004 Slyunix.tar.bz2
drwxr-xr-x 2 seb users 4096 may 13 2006 snap
drwxr-xr-x 2 seb users 4096 may 13 2006 Solutions_Linux_2005
drwxr-xr-x 2 seb users 4096 may 13 2006 statuts_declaration
-rw-r--r-- 1 seb users 10217 dic  5 2004 statuts.sxw
-rwrxr-xr-x 1 seb users 146687 mar 29 2006 suselinux101.odp
-rw-r----- 1 seb users 68096 jul 31 2004 Thumbs.db
drwxr-xr-x 3 seb users 4096 may 13 2006 t-shirt
-rw-r--r-- 1 seb users 41775 ago 29 2004 xev.jpg
seb@slyserver:~/Documents/slyunix>

```

Xterm, el clásico emulador de terminal

Para configurar Xterm, tiene que utilizar los botones del ratón y la tecla [Ctrl]:

- [Ctrl] + botón izquierdo: opciones principales;
- [Ctrl] + botón derecho: elección de las fuentes de caracteres;
- [Ctrl] + botón del medio: opciones del terminal.

Existen otros emuladores de terminales que podrían gustarle:

- **rxvt**: un terminal muy ligero, pequeñito, que se parece a Xterm pero sin sus menús de configuración. Tenga en cuenta que se verá obligado a utilizar una versión especial llamada urxvt si su distribución es UNICODE.
- **aterm**: un terminal cercano a rxvt, previsto para el gestor Afterstep, que soporta la seudotransparencia del fondo.
- **eterm**: un terminal previsto para sustituir a xterm (como todos los terminales, de hecho) y que fue uno de los primeros en proponer menús. Soporta pestañas en sus últimas versiones.
- **konsole**: el terminal entregado con KDE: soporta perfiles, varios fondos, estilos, transparencia y pestañas. No es el más ligero en tamaño, pero sí uno de los menos voraces en recursos y de los más rápidos.
- **gnome-terminal**: el terminal entregado con GNOME: propone lo mismo que el de KDE, ¿o será al revés?...
- etc.

Puede elegir el terminal que desee, según sus gustos y sea cual sea su entorno de trabajo. Lo importante es lo que hace usted con él: ejecutar comandos, crear y ejecutar scripts, etc. Sólo recuerde que iniciar un terminal o cualquier otro programa del entorno gráfico tiene como consecuencia consumir recursos, entre otros la memoria. Por ejemplo, iniciar konsole desde GNOME, que, por cierto, funciona muy bien, supone cargar en memoria la ristra de componentes de los que konsole depende.

## 4. Los gestores de ventanas

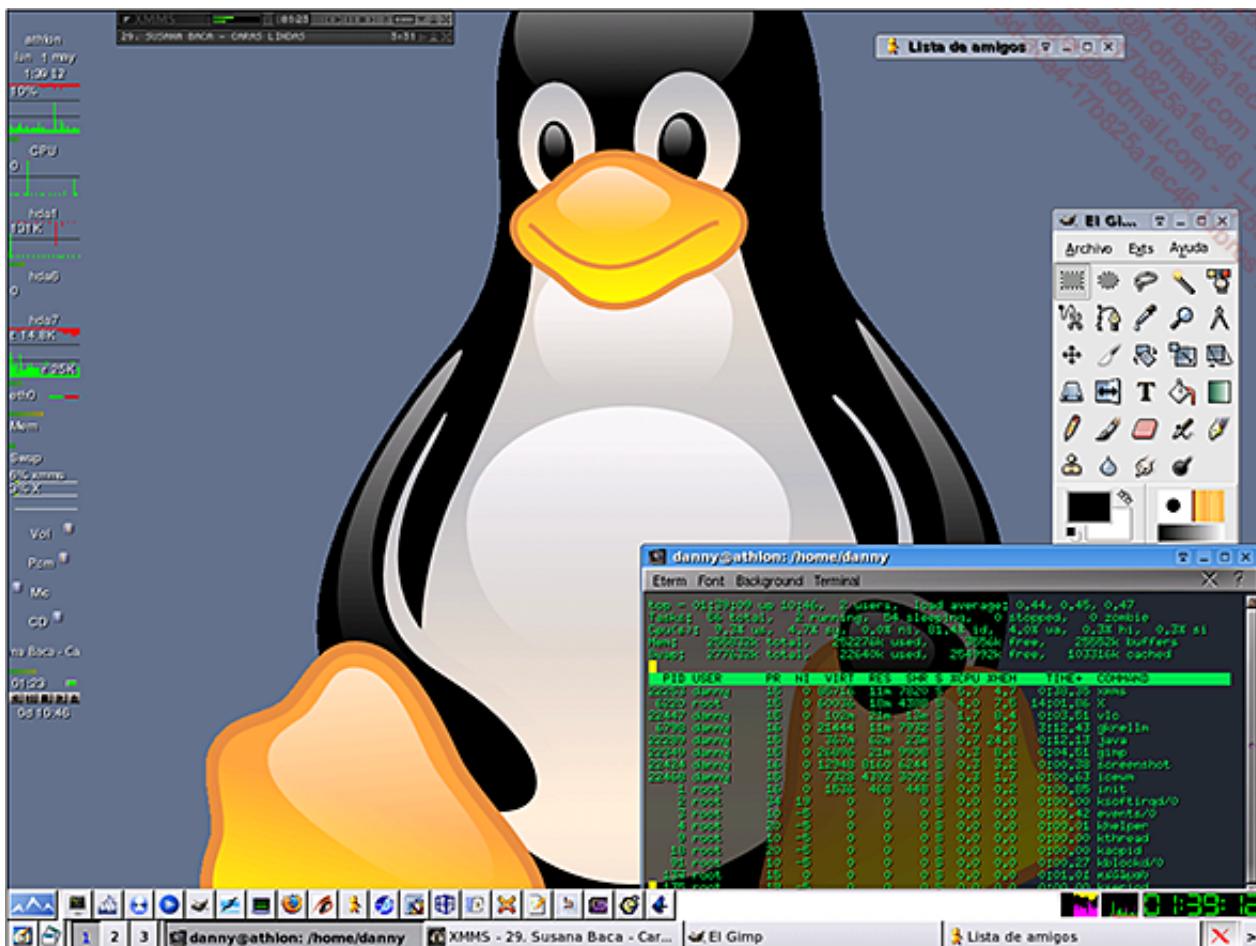
Se ha tratado el principio de los gestores de ventana al comienzo del capítulo. Puede elegir un entorno según sus gustos y necesidades. A continuación le damos una lista de los gestores de ventanas y entornos de escritorio más famosos.

### a. twm

El único gestor de ventana oficial de X Window desde la versión X11R4 es TWM: Tom's Window Manager. Es el único que se entrega por defecto, sea cual sea la versión de X. En su versión básica, tal y como pudo ver en la captura al principio de este capítulo, twm es mínimo y está completamente superado por los últimos entornos. Sin embargo, es extremadamente configurable, suministra barras de títulos, iconos, etc. Su configuración se basa en ficheros de textos.

### b. IceWM

Creado para ser agradable a nivel visual y muy ligero, IceWM presenta la ventaja de sus numerosos temas gráficos, que permiten que se parezca a otros entornos como Windows, OS/2 o lo que quiera. En particular se utiliza IceWM por defecto con los eeePC de Asus (los que se ejecutan en Linux). Su configuración utiliza ficheros de texto, aunque cuenta con varios paneles de controles.



*IceWM. Captura con licencia GPL procedente de Wikimedia Commons*

### c. fvwm

fvwm es un derivado de twm. La última versión oficial, la 1.24, fue publicada en 1994, y desde entonces han ido apareciendo otras versiones más evolucionadas, como la fvwm95, que se caracterizaba por imitar la barra de las tareas y la estética de las ventanas de Windows 95 y Windows 98. Es altamente configurable y está en la génesis de otros gestores como Afterstep (previsto para agrupar a NextStep), Xfce y Enlightenment. Bien configurado (ficheros de texto y aplicaciones gráficas), es espléndido y ligero. Todavía se utiliza hoy en día, y es facilitado y soportado por la mayoría de los editores de distribuciones.

#### **d. CDE**

CDE (*Common Desktop Environment*) es un entorno de escritorio basado en Motif y procedente en su origen de openVMS, de HP. Dado que este entorno fue propuesto por The Open Group como estándar, hubiera podido convertirse en común a todos los Unix (de hecho, Solaris, HPUX y True64 lo instalan por defecto). Pero al ser software propietario y haberse liberado Motif demasiado tarde, las plataformas libres se volcaron en los demás, en particular KDE y GNOME.

#### **e. WindowMaker**

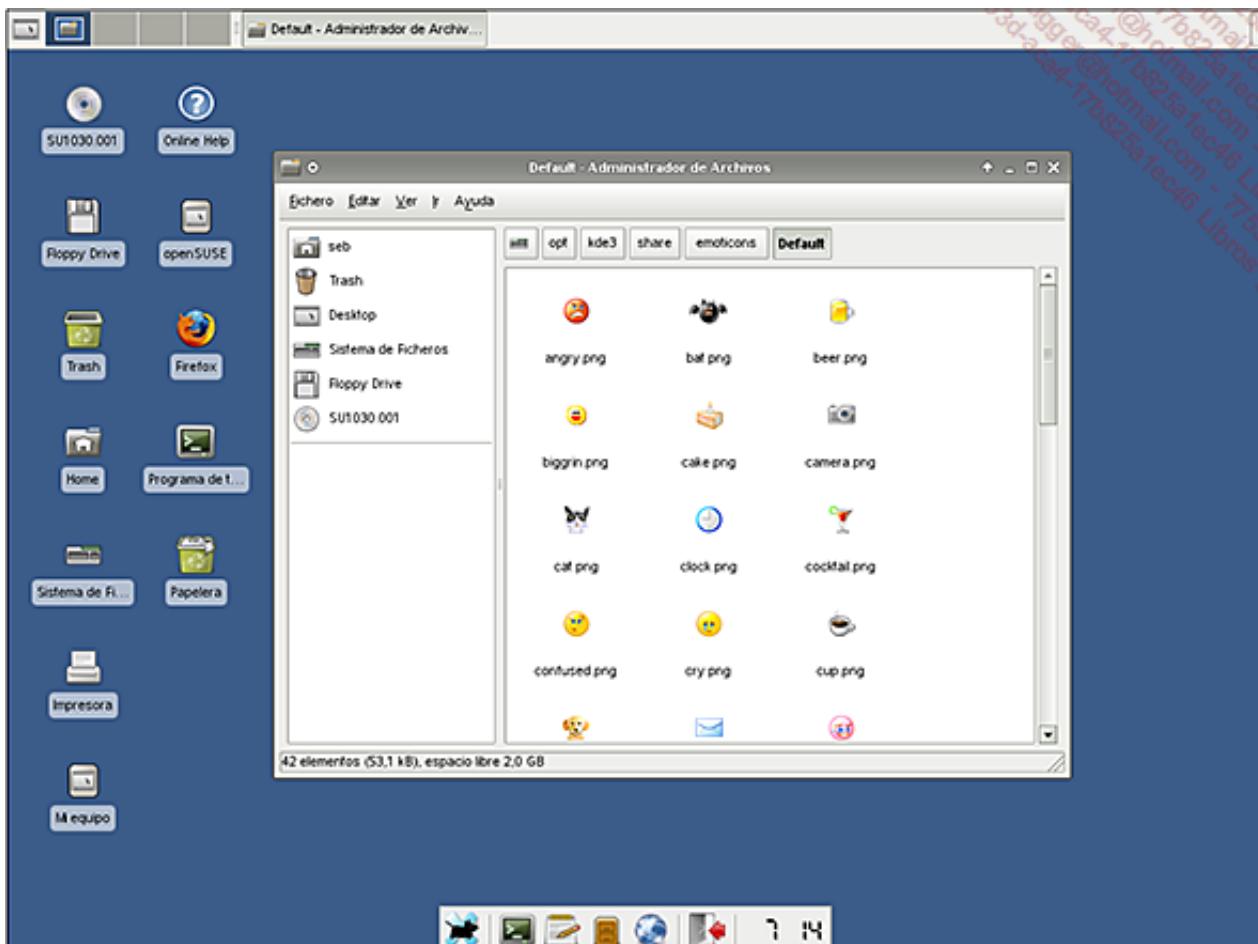
WindowMaker, del que mostramos una captura al principio de este capítulo, es un gestor de ventanas muy evolucionado, desarrollado para parecerse a Nextstep y funcionar con GNUstep, el entorno compatible OpenSTEP de las máquinas NeXT de Steve Jobs. Originalmente se trataba de una versión mejorada de AfterStep. Es estable, fiable, muy bien acabado, fácil de configurar y relativamente ligero.

#### **f. Enlightenment**

Enlightenment es el gestor de ventanas de referencia para numerosos geeks y nerds. Es totalmente configurable, soporta la ejecución de scripts y es modular. Funciona tanto en pequeñas configuraciones como en grandes servidores. La versión actual, la 16 (actualmente 16.999) existe desde hace varios años, y la versión 17 es la eterna promesa, o el «vaporware» del software libre, con la diferencia de que saldrá realmente un día: cuando esté lista. Será entonces un verdadero entorno de escritorio (Desktop Shell).

#### **g. Xfce**

Frente a los mastodónticos KDE y GNOME, las máquinas que disponen de pocos recursos tienen la alternativa Xfce. Fue desarrollado desde esta óptica: un entorno de escritorio completo y muy ligero, basado en las librerías GTK, pero sin consumir recursos de manera excesiva. Funciona, por lo tanto, muy rápidamente con 64 MB de memoria y se entrega con varias aplicaciones, entre las cuales se cuentan un gestor de ficheros (Thunar), un editor de texto, etc. En resumen: lo mínimo para trabajar. El usuario cuenta con muchas aplicaciones disponibles para este entorno.



*El entorno de escritorio XFCE*

## **h. KDE y GNOME**

KDE y GNOME: se presentan al principio de capítulo. La elección es excelente si su máquina dispone de al menos 256 MB (512 para funcionar cómodamente) y si desea tener un entorno comparable con el de Windows y Mac OS.

## **i. Los demás**

Existen decenas de otros gestores de ventanas, entre los cuales se cuentan, por ejemplo, Compiz-Fusion, que representa probablemente el futuro con sus efectos 3D espléndidos y compuestos. Entre los gestores de ventanas notables encontrará:

- Afterstep
- Compiz Fusion
- Fluxbox
- Openbox
- Metacity
- Blackbox
- Ion

- Wmii
- etc.

## 5. Exportar las ventanas

Como pudo ver durante su presentación, X funciona en modo cliente-servidor. Por defecto, el cliente se conecta al servidor X local y se visualiza en la pantalla principal del usuario. Sin embargo, teóricamente se puede exportar la visualización hacia cualquier servidor X.

Supongamos dos servidores X:

- uno sobre 192.168.1.60;
- otro sobre 192.168.1.70.

Usted quiere iniciar desde el primero el programa **xcalc** hacia el segundo.

Para ello, debe autorizar al segundo servidor X a recibir peticiones exteriores:

- la opción **-nolisten tcp** debe estar deshabitada mediante X o XDM;
- tiene que autorizar la conexión con el comando **xhost**.

El comando **xhost** permite controlar los permisos de acceso al servidor X. Su sintaxis básica toma un más «+» o un menos «-» para autorizar o no de manera global (todo el mundo) las conexiones al servidor X.

```
$ xhost +
access control disabled, clients can connect from any host
$ xhost -
access control enabled, only authorized clients can connect
```

Después de los signos, puede precisar un nombre de anfitrión o de login. En el segundo servidor X, debe autorizar el primero:

```
$ xhost +192.168.1.60
192.168.1.60 being added to access control list
```

Desde la primera máquina tiene dos maneras de exportar la visualización hacia la segunda:

- solamente el programa con el parámetro **-display**;
- todos los programas exportando una nueva variable **DISPLAY**.

En los dos casos tiene que precisar, además de la dirección, el número de servidor y si es preciso de pantalla. Se asigna un identificador a cada servidor X. Si es el único, es el cero (0); si es el segundo, 1, y así sucesivamente. Las «coordenadas» del servidor X, si está solo, son **192.168.1.70:0**. Si se han acoplado varias pantallas, se identifican de manera numérica: 0,0,1,2, etc. Si no se precisa ninguna pantalla, se elige la primera por defecto. Por lo tanto una coordenada completa es **192.168.1.70:0.0**.

En el primer caso proceda como a continuación:

```
$ xcalc --display 192.168.1.70:0
```

En el segundo caso, exporte la variable **DISPLAY** con el nuevo valor:

```
$ export DISPLAY=192.168.1.70:0$
```

 La política de seguridad por defecto consiste en impedir el acceso al servidor X incluso si se precisa xhost+. En este caso, y esperando que lo permita el servidor ssh, pase por ssh con el parámetro -X.

---

## 6. Los recursos de una aplicación X

### a. Modificar la apariencia de un programa

Cada programa X invoca recursos del servidor X Window para poder funcionar. En concreto, necesita del servidor las fuentes de caracteres, la forma de los botones, los colores, etc. El fichero Xresources de la configuración de Xdm es un buen ejemplo.

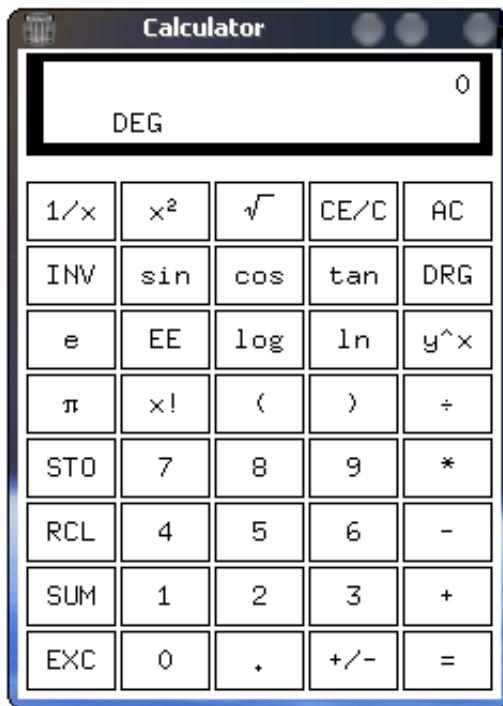
Es posible establecer los recursos que puede utilizar un programa X, excepto con las aplicaciones KDE o GNOME, que no están sujetas a dicha limitación.

Cada programa X admite varios parámetros estándares en la línea de comandos, entre los cuales se encuentran:

- **-bg**: color de fondo, recurso \*background.
- **-bd**: color de borde, recurso \*borderColor.
- **-fg**: color de primer plano, recurso \*foreground.
- **-fn**: fuente principal, recurso \*font.
- **-geometry**: tamaño y posición de la ventana, recurso \*TopLevelShell.geometry.
- **-title**: título de la ventana, recurso \*title.
- etc.

Xcalc puede iniciarse en dos modos: TI-30 y HP (notación polaca invertida).

En el ejemplo siguiente se propone modificar la visualización de la aplicación **xcalc**. Primero, una captura de la calculadora antes de las modificaciones.



*La calculadora X por defecto*

Y ahora tenemos el mismo resultado con un cambio de los parámetros por defecto: título «Calculadora», fondo gris, color principal marrón, bordes amarillos:

```
$ xcalc -title "Calculadora" -bg grey -fg brown -bd yellow &
```



*La misma calculadora, modificada mediante la línea de comandos*

## b. Editar los recursos

El comando **xprop** permite conocer todas las propiedades de una ventana X.

- Primero inicie xcalc.

```
$ xcalc &
```

- Luego inicie xprop desde un terminal.

```
$ xprop
```

- El cursor del ratón se transforma en una cruz. Haga clic en la ventana de xcalc. Tendría que obtener una salida de este tipo:

```
NET_WM_ICON_GEOMETRY (CARDINAL) = 649, 984, 133, 20
WM_PROTOCOLS(ATOM): protocols WM_DELETE_WINDOW
WM_STATE(WM_STATE):
    window state: Normal
    icon window: 0x0
_NET_WM_STATE(ATOM) =
_NET_WM_ALLOWED_ACTIONS(ATOM) = _NET_WM_ACTION_MOVE,
_NET_WM_ACTION_RESIZE, _NET_WM_ACTION_MINIMIZE,
_NET_WM_ACTION_SHADE, _NET_WM_ACTION_MAXIMIZE_VERT,
_NET_WM_ACTION_MAXIMIZE_HORZ, _NET_WM_ACTION_FULLSCREEN,
_NET_WM_ACTION_CHANGE_DESKTOP, _NET_WM_ACTION_CLOSE
KDE_NET_WM_FRAME_STRUT(CARDINAL) = 5, 5, 19, 5
_NET_FRAME_EXTENTS(CARDINAL) = 5, 5, 19, 5
_NET_WM_DESKTOP(CARDINAL) = 0
WM_CLIENT_LEADER(WINDOW): window id # 0x3e00026
WM_LOCALE_NAME(STRING) = "es_ES.UTF-8"
KDE_NET_WM_USER_CREATION_TIME(CARDINAL) = 1321633138
WM_CLASS(STRING) = "xcalc", "XCalc"
"WM_HINTS(WM_HINTS):
    Client accepts input or input focus: True
    Initial state is Normal State.
    bitmap id # to use for icon: 0x3e00001
WM_NORMAL_HINTS(WM_SIZE_HINTS):
    program specified size: 226 by 308
    window gravity: NorthWest
WM_CLIENT_MACHINE(STRING) = "slyserver"
WM_COMMAND(STRING) = { "xcalc" }
WM_ICON_NAME(STRING) = "Calc"
WM_NAME(STRING) = "Calculator"
```

En la información devuelta, la línea en negrita indica el nombre (xcalc o XCalc) que se deberá utilizar para acceder a los recursos de la aplicación X. Suele ser el nombre real del programa.

Un recurso se presenta con la forma:

```
class*resource: value
instance*resource: value
class.resource: value
instance.resource: value
```

**class** representa todas las invocaciones de un mismo programa. La clase XCalc representa todas las instancias de XCalc, por lo tanto si usted ejecuta n veces el programa xcalc los recursos de tipo xcalc.xxxx serán comunes a todos los xcalc iniciados.

- **instancia** es el nombre de una instancia de un programa particular.

El manual de xcalc facilita en las secciones CUSTOMIZATION y WIDGET HIERARCHY los recursos y valores particulares aceptados. Xcalc dispone de los modos hp y ti. En modo ti, por ejemplo, los recursos son del

tipo:

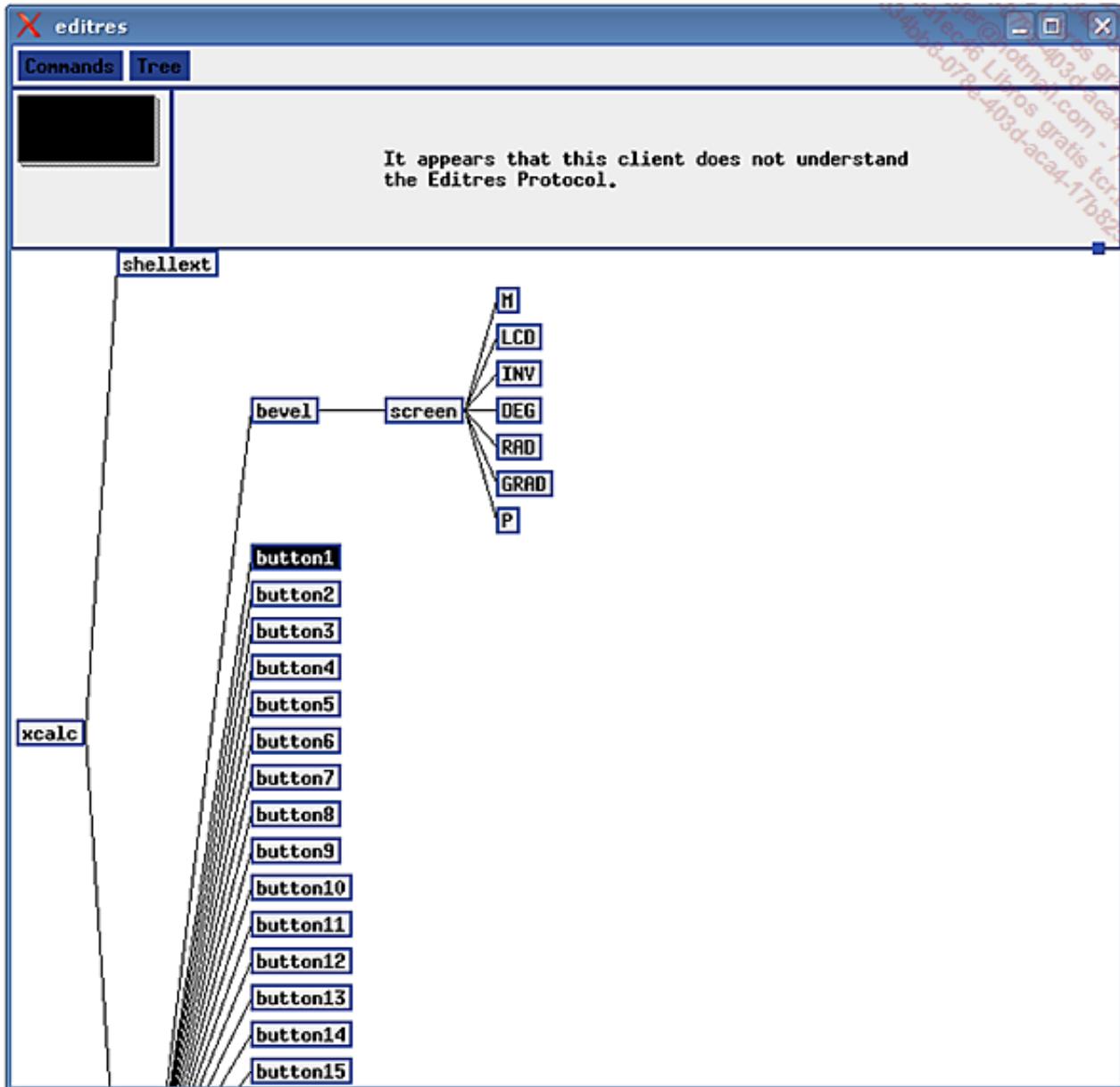
```
XCalc.ti.resource: value
```

Pero si sustituye ti o hp por un asterisco «\*», la definición del recurso se aplicará al conjunto. Los propios recursos se pueden descomponer en subrecursos.

El comando **appres** toma como parámetro una instancia o una clase y devuelve todos los recursos asociados. Es ideal para guardar los recursos de una clase ante de modificarlos. Sin embargo, visualiza también los recursos aplicables a XCalc, que por defecto empiezan por «\*»:

```
$ appres XCalc | grep -i xcalc
...
XCalc*ti.button13.fromVerde: button8
XCalc*ti.button30.translations: #override
<Btn1Down>,<Btn1Up>:subtract()unset()
XCalc*ti.button30.background: rgb:e/d/c
XCalc*ti.button30.borderColor: rgb:9/8/7
XCalc*ti.button30.foreground: gray5
XCalc*ti.button30.fromVerde: button25
XCalc*ti.button30.fromHoriz: button29
XCalc*ti.button30.label: -
XCalc*ti.button30.displayList: foreground rgb:a/9/8;segments
8,-4,-9,-4,-4,-9,-4,8;draw-arc -14,-14,-4,-4,270,90
XCalc*ti.button14.label: ln
XCalc*ti.button14.translations: #override
<Btn1Down>,<Btn1Up>:naturalLog()unset()
XCalc*ti.button14.fromHoriz: button13
XCalc*ti.button14.fromVerde: button9
XCalc*ti.Command.background: rgb:c/d/e
XCalc*ti.Command.borderColor: rgb:8/9/a
XCalc*ti.Command.shapeStyle: roundedRectangle
XCalc*ti.Command.foreground: gray5
XCalc*ti.Command.displayList: foreground rgb:a/b/c;segments
8,-4,-9,-4,-4,-9,-4,8;draw-arc -14,-14,-4,-4,270,90
XCalc*ti.backgroundPixmap:
gray3?foreground=gray70&background=gray85
XCalc*bevel.screen.GRAD.fromHoriz: RAD
XCalc*bevel.screen.GRAD.fromVerde: LCD
XCalc*bevel.screen.INV.fromVerde: LCD
XCalc*bevel.screen.INV.verdeDistance: 4
XCalc*bevel.screen*INV.verdeDistance: 2
XCalc*bevel.screen.P.label: ()
XCalc*bevel.screen.P.fromHoriz: GRAD
XCalc*bevel.screen.P.fromVerde: LCD
XCalc*bevel.screen.P.horizDistance: 2
XCalc*bevel.screen.LCD.label: 88888888888
XCalc*bevel.screen.LCD.fromHoriz: M
XCalc*bevel.screen.LCD.horizDistance: 4
XCalc*bevel.screen.LCD.verdeDistance: 2
XCalc*bevel.screen*LCD.foreground: gray20
XCalc*bevel.screen.DEG.fromHoriz: INV
XCalc*bevel.screen.DEG.fromVerde: LCD
XCalc*bevel.screen.DEG.horizDistance: 1
XCalc*bevel.screen.Label.horizDistance: 4
XCalc*bevel.screen.Label.verdeDistance: 2
XCalc*bevel.screen.Label.internalHeight:1
XCalc*bevel.screen.Label.internalWidth: 1...
```

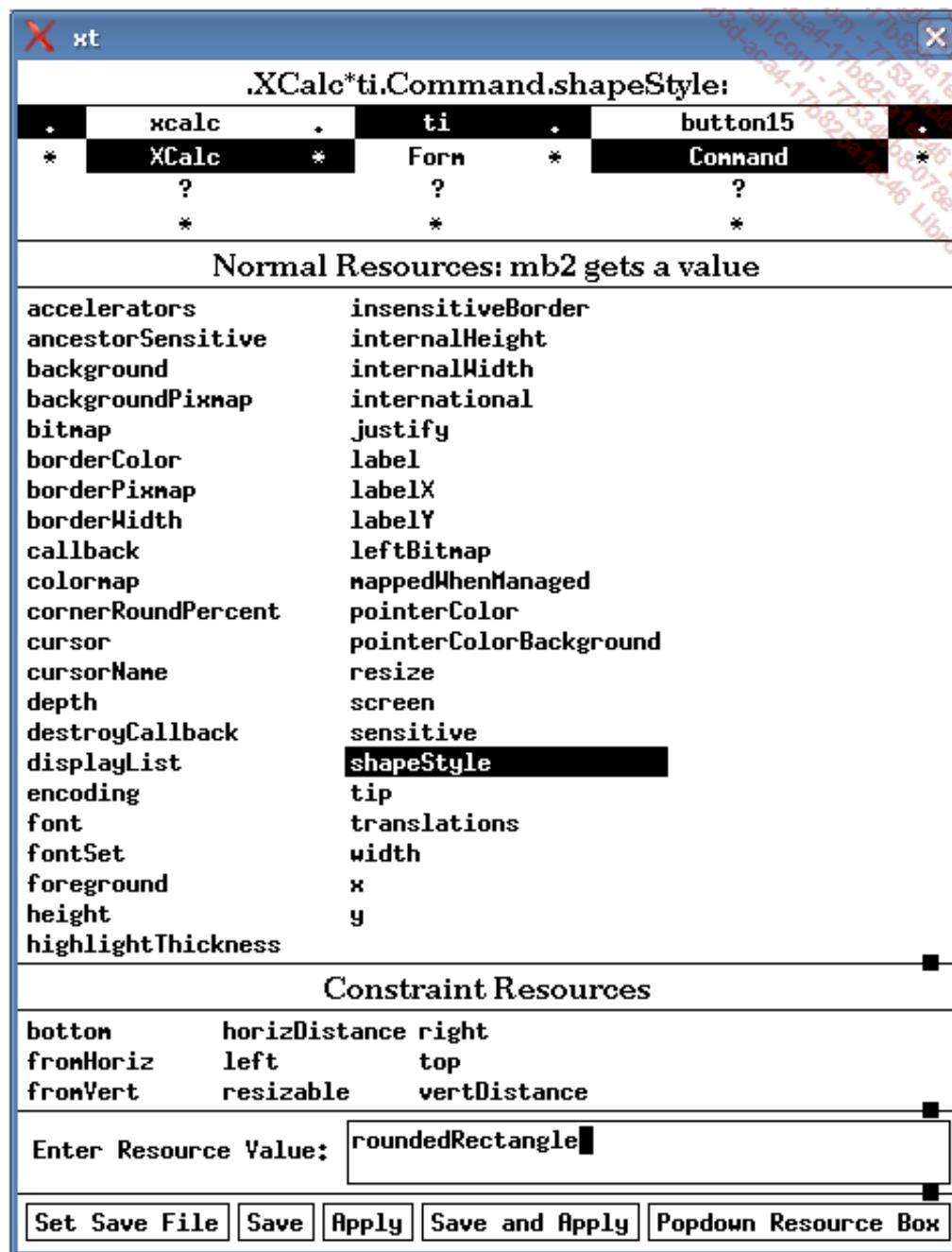
Existe un editor simple para editar los recursos de un programa y guardar los parámetros. Se llama **editres**. Inícielo desde el menú **Commands**, seleccione **Get Tree**. Luego haga clic en la ventana X, la de Xcalc.



*editres: editor de recursos X*

Para modificar un valor a voleo, haga clic en un recurso, por ejemplo **button1**, luego en **Commands**sobre **Show Resource Box**. En este cuadro elija **background**, teclee **red** y luego haga clic en **Apply**. El fondo del botón **1/x** se vuelve rojo.

En el menú **Tree**, seleccione **Show class name**. Haga clic en uno de los **Command** y luego vuelva al cuadro de los recursos. Va a modificar todos los recursos para todas las clases **Command**: haga clic arriba en **XCalc**, **\***, **ti**, **.**, **Command**, luego en el cuadro central en **shapeStyle** (arriba debe aparecer **XCalc\*ti.Command.shapeStyle**). Teclee **roundedRectangle**, luego **Apply**. ¡Los botones son todos redondeados!



Edición de un recurso de xcalc con editres

Puede guardar el ajuste con los botones **Set Save File** y luego **Save**.

```
$ cat Xcalc.res
.XCalc*ti.Command.shapeStyle: roundedRectangle
```

### c. xrdb

Puede guardar sus ajustes en un fichero cualquiera, pero X intenta leer el fichero \$HOME/.Xdefaults en el momento de abrir la sesión. Se utiliza el comando **xrdb** para editarlo. Admite los parámetros siguientes:

- **ninguno**: el nuevo fichero sobreescribe todos los antiguos ajustes;
- **-merge**: se añade el fichero a los ajustes anteriores;

- **-remove**: suprime un recurso. Sin nada: suprime todos los recursos.

Supongamos el contenido siguiente:

```
$ cat .Xdefaults
XCalc*title: Calculadora
XCalc*ti.Command.shapeStyle: roundedRectangle
XCalc*ti.Command.background: cyan
```

título de la ventana será Calculadora.

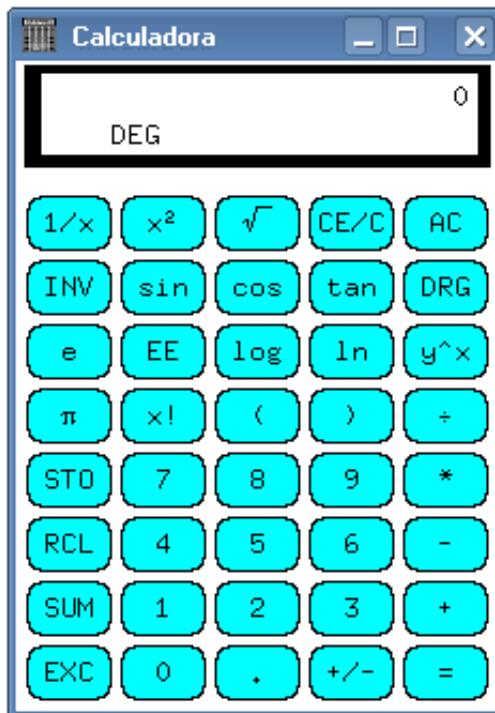
- Los botones tendrán las esquinas redondeadas.
- El fondo de los comandos (los botones) serán azul celeste.

Cargue los ajustes como a continuación:

```
$ xrdb -merge .Xdefaults
```

Y mire el resultado:

```
$ xcalc
```



*xcalc modificado mediante el fichero de los recursos.*

# Accesibilidad

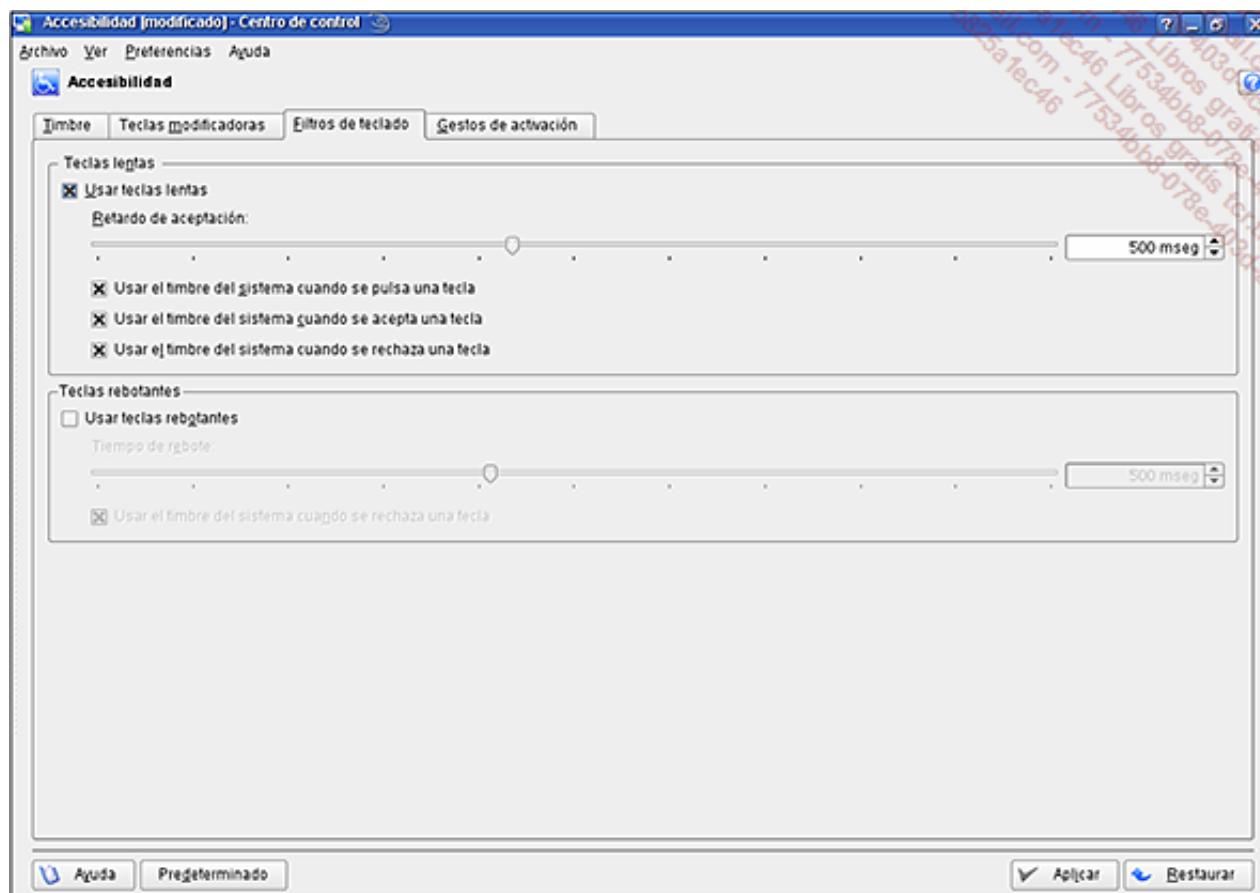
Si bien la mayoría de los usuarios no tienen problemas para leer una pantalla, escribir con un teclado y utilizar un ratón, otros pueden padecer trastornos que les impidan trabajar de manera normal. Según la minusvalía, Linux y sus herramientas proponen varios medios para ayudar a estas personas.

Esta sección repasa algunos de los medios disponibles para ayudar a los minusválidos, en particular con discapacidades motrices o visuales.

## 1. Asistencia al teclado y el ratón

Un simple ajuste del teclado puede ayudar a una persona con problemas motores. Por ejemplo, una persona lenta va a dedicar mucho tiempo a buscar una tecla, con el riesgo de equivocarse; luego apretará demasiado tiempo la tecla deseada, lo que tendrá como efecto repetir el carácter tecleado varias veces.

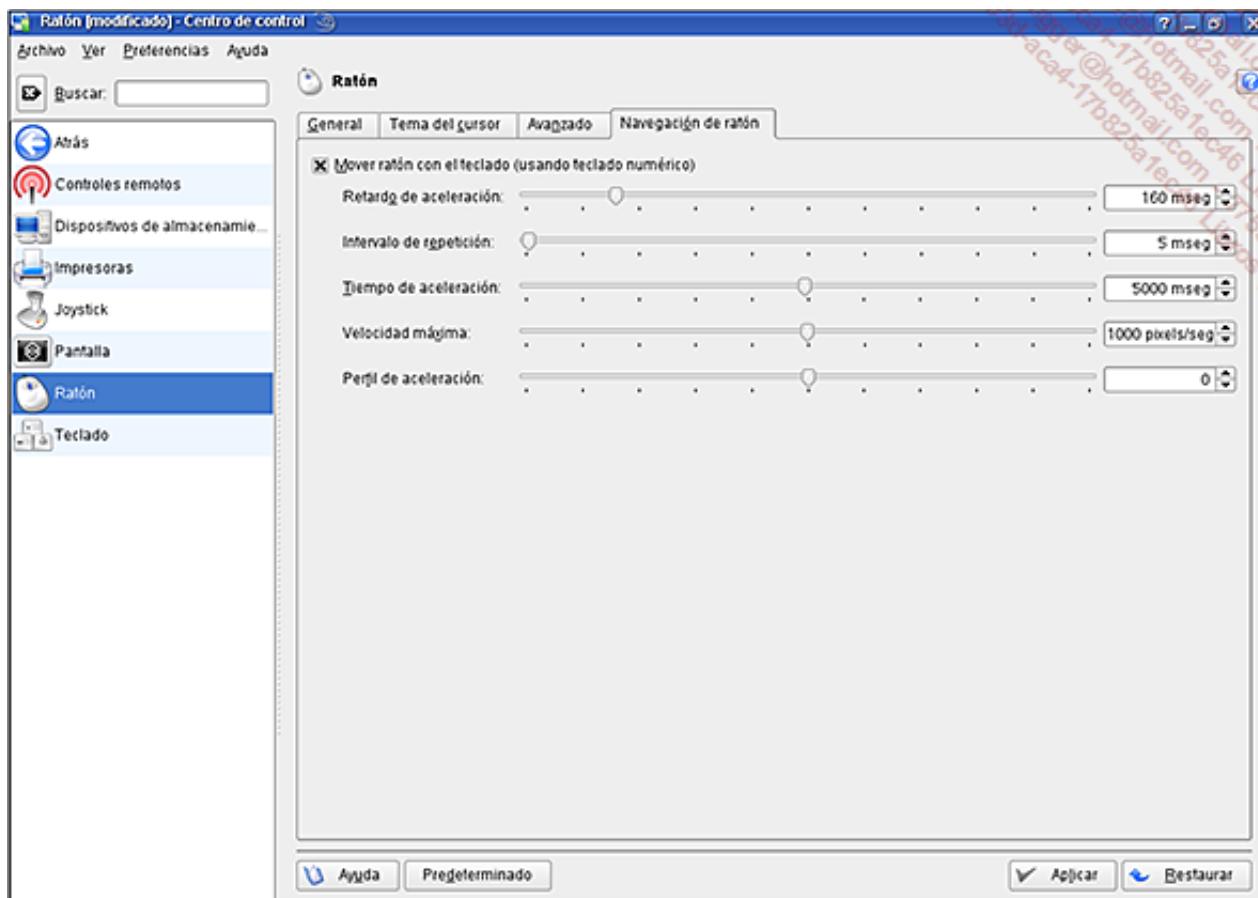
El módulo AccessX del entorno gráfico permite muchos ajustes. Son accesibles desde los centros de configuración de los entornos gráficos. En KDE4, por ejemplo, el módulo **Accesibilidad** de la pestaña **General** de la configuración del sistema propone modificar numerosos ajustes por defecto, entre los cuales se encuentra el funcionamiento del teclado. Por ejemplo, en la pestaña **Filtros de teclado**, marque **Usar teclas lentas**. Si una persona quiere efectuar una acción, no se tendrá en cuenta hasta que se cumpla el plazo elegido (0,5 segundos por defecto). Debajo, **Usar teclas rebotantes** espera el plazo indicado antes de repetir una tecla en caso de apretar durante demasiado tiempo.



Los parámetros de accesibilidad para el teclado de KDE4

En el mismo sentido, puede activar un soporte visual o auditivo según las acciones efectuadas. Al presionar una tecla puede provocar un bip, el parpadeo de la pantalla. Los movimientos del ratón pueden contribuir a la mejora de la accesibilidad.

Como no todo el mundo puede utilizar el ratón, es posible desplazar el cursor en la pantalla usando el teclado. Lea la documentación de su entorno para la activación de este modo operativo. Siempre en KDE4, en el módulo **Teclado y Ratón**, pestaña **Navegación de ratón**, si marca **Mover ratón con el teclado**, podrá mover el cursor del ratón con las teclas direccionales del teclado numérico. Las teclas de 1 a 9, salvo la 5, desplazan el cursor en la dirección deseada. La tecla 5 simula un simple clic (5 dos veces rápidamente simula un doble clic). Es posible cambiar de botón, mantenerlo, etc.



#### *Utilización de las teclas del teclado numérico para desplazar el ratón*

En lo que se refiere al ratón, si usted dispone de un modelo correcto, puede activar el perfil para zurdos, que va a invertir los botones e incluso el sentido de la rueda.

Finalmente, el teclado virtual puede proporcionar una ayuda notable en caso de dificultad de utilización del teclado físico: a partir de entonces, puede utilizar un teclado de software con el ratón o una pantalla táctil. Basta con hacer clic en las diferentes teclas para utilizarlo como un teclado ordinario. El de KDE se llama **kvkbd**, pero X Window propone uno por defecto llamado **xvkbdb**, y Gnome propone GOK.



*El teclado virtual kvkbd permite pasar sin el teclado físico.*

## 2. Asistencia visual y auditiva

Las personas con problemas de visión apreciarán la posibilidad de utilizar notificaciones visuales, como el parpadeo de la pantalla. Además, todos los entornos de escritorio dignos de este nombre, como KDE3, KDE4 o GNOME (por ejemplo), proponen modificar la visualización:

- **Colores:** aumentando por ejemplo los contrastes o con temas según la representación de los colores (daltonismo).
- **Estilos:** modificando el tamaño de varios elementos visuales, como los botones, las casillas, los campos, etc.
- **Fuentes de caracteres:** elegiendo una fuente adaptada, de mayor tamaño, de un estilo determinado.
- **Temas:** existen temas específicos para los invidentes que modifican el conjunto de los ajustes anteriores de una vez.

Hay herramientas sencillas que permiten ampliar zonas de la pantalla con una lupa virtual. KDE propone Kmagnifier. También es posible asociar acciones (teclas del teclado o del ratón) a la lupa para tener un acceso más rápido.

Las personas invidentes pueden también trabajar con Linux, ya que existen soluciones para leer una pantalla en braille. Existen varias soluciones que soportan el braille, entre otras brltty, para el traspaso de la consola a un lector de braille, y orca, producto estrella en Gnome, que proporciona el soporte del braille en el entorno Gnome, pero también voces sintéticas y la accesibilidad a la pantalla para las personas ciegas.

Para terminar, existen varios medios para «hacer hablar» al ordenador. Productos tales como espeak (sintetizador muy «computerizado») y sobre todo mbrola con freetts dan resultados muy cercanos a una voz humana.

# **Validación de los conocimientos adquiridos: preguntas/respuestas**

## **1. Preguntas**

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las siguientes preguntas:

### **X Window**

**1** ¿Qué componente de X Window permite comunicarse a los clientes y los servidores X?

- A - Xt
- B - Motif
- C - Xlib
- D - Qt

**2** ¿Es posible visualizar en una segunda máquina una aplicación X iniciada desde una primera?

**3** ¿Quién define el modelo y el estilo de las ventanas?

**4** ¿Es posible elegir el gestor de ventanas?

**5** ¿Qué es un Widget?

- A - Una función de gestión de ventanas.
- B - Un elemento básico de la interfaz gráfica.
- C - Un estilo de visualización.
- D - Una librería de gestión de los eventos.

**6** Cite al menos dos toolkits diferentes y habituales.

**7** ¿Qué puede hacer si hay demasiadas ventanas abiertas en su escritorio y no consigue aclararse?

**8** ¿Qué proyecto intenta federar las bases de la integración de los diferentes proyectos relacionados con el entorno de escritorio de Linux?

- A - El proyecto Xfce.
- B - El proyecto KDE.
- C - El proyecto GNOME.
- D - El proyecto Freedesktop.

### **Xorg**

**9** ¿De qué proyecto deriva X.org?

**10** ¿Dónde se encuentra el fichero de configuración de Xorg?

**11** ¿Cuál es el mejor método para instalar Xorg?

**12** ¿Cuáles son los cuatro medios disponibles para configurar X Window presentados en este libro?

**13** ¿Cuál es la sección del fichero de configuración de Xorg que permite configurar un teclado o un ratón?

- A - InputDevice
- B - Mouse
- C - Keyboard

- D - Modules

**14** ¿Qué opción relacionada con el driver kbd permite modificar el idioma del teclado?

**15** Su ratón dispone de una rueda constituida por la combinación de los botones 4 y 5. ¿Qué opción debe añadir en la sección correspondiente de Xorg?

**16** ¿Es obligatorio especificar las frecuencias del monitor?

**17** Su monitor soporta una resolución que Xorg no reconoce. ¿Qué posibilidad tiene para «forzar» a Xorg a utilizar esta resolución?

**18** La zona de visualización es demasiado ancha y los botones de ajustes de la pantalla no arreglan nada. ¿Qué herramienta puede ayudarle a rectificar la anchura de la visualización?

**19** ¿De qué dependen las opciones de la sección Device?

**20** ¿Cuál es el papel de la sección Screen?

- A - Describe el monitor.
- B - Describe la tarjeta gráfica.
- C - Describe los parámetros de visualización.
- D - Describe la configuración global del servidor X.

**21** Su servidor X debe arrancar en 1.600x1.200 por defecto y proponer las resoluciones 1.280x1.024 y 1.024x768. ¿Cuál es la línea asociada a «Modos»?

**22** ¿Cuál es el papel de la sección ServerLayout?

**23** No quiere llenar las rutas FontPath del fichero de configuración X. ¿Dónde colocará sus fuentes para que sean incorporadas automáticamente?

**24** ¿Cuáles son los tres módulos que, en teoría, deben cargarse en memoria para activar las extensiones OpenGL y el soporte de hardware 3D?

**25** ¿Cómo impedir que un usuario desconfigure su servidor X?

**26** El inicio de X Window ha sido tan directo que devuelve un error relacionado con su parámetro -probeonly. ¿Qué tiene que hacer?

**27** El inicio de X lleva a la visualización de una pantalla gris con una cruz en medio y nada más. ¿Qué está pasando?

## **Los gestores de visualización XDM**

**28** ¿Para qué sirve XDM?

- A - Abrir una sesión gráfica local.
- B - Abrir una sesión gráfica remota.
- C - Autenticar a un usuario.
- D - Gestionar múltiples sesiones X.

**29** ¿Cuál es el nombre del protocolo X que permite gestionar las conexiones y aperturas de sesiones remotas?

**30** ¿Es posible iniciar KDE desde GDM?

**31** Señale algunas diferencias entre los Display Manager de KDE o GNOME y xdm.

**32** Dé la ruta (directorio) de los ficheros de configuración de xdm.

**33** ¿Cómo se llaman los parámetros que permiten ajustar xdm?

- A - variables,
- B - argumentos,

- C - recursos,
- D - parámetros.

**34** ¿Cuál es el contenido principal del fichero xdm-config?

**35** ¿En qué momento se ejecuta el comando Xsetup?

**36** ¿Cuál es el primer fichero que se ejecuta en el entorno del usuario después de su conexión?

**37** ¿Dónde puede colocar su propia configuración de sesión el usuario común?

- A - ~/.xinitrc
- B - ~/.xsession
- C - ~/.xstartup
- D - ~/.xresources

**38** ¿Cómo asegurarse de encontrar el fichero de configuración de kdm o gdm?

**39** ¿Cuál es el nivel de ejecución para arrancar en modo gráfico en la mayoría de las distribuciones?

- A - 3
- B - 4
- C - 5
- D - Irrelevante: xdm se lanza mediante un script en /etc/init.d

**40** ¿Cómo determina gdm o kdm qué tipo de sesión por defecto se debe iniciar al conectarse un usuario determinado?

- A - Por el contenido del directorio /usr/share/xsessions.
- B - Por el contenido del fichero /etc/sysconfig/desktop.
- C - Por la variable del sistema DISPLAYMANAGER.
- D - por la lectura del contenido del fichero ~/.dmrc.

**41** ¿Qué herramienta sustituye la consola en X?

**42** ¿Cuántos gestores de ventanas hay?

- A - 4
- B - 7
- C - 15
- D - difícil de saber.

**43** ¿Cómo exportar la visualización de todos los programas X en la primera pantalla del primer servidor X 10.17.32.142?

## **Recursos X**

**44** ¿Cómo iniciar xcalc con las teclas en fondo marrón (brown)?

**45** ¿Cómo conocer la «clase» de una ventana X para luego poder modificar sus recursos?

**46** ¿Cómo colocar en un fichero xclock.ressources el conjunto de los recursos del proceso de clase xclock (el comando xclock)?

**47** ¿Cómo modificar de manera dinámica y gráfica los recursos de una aplicación X iniciada?

**48** El recurso que permite modificar el color de la aguja de las horas en xclock es «xclock.clock.hourColor». ¿Cómo lograr que a partir de las próximas conexiones y de manera definitiva el color de la aguja sea negro?

## 2. Resultados

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /48

Para este capítulo, su resultado mínimo debe ser de 36 respuestas acertadas (36/48).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- X Window.
- Xorg.
- Los gestores de visualización XDM.
- Recursos X.

## 3. Respuestas

### X Window

**1** ¿Qué componente de X Window permite comunicarse a los clientes y los servidores X?

- A - Xt
- B - Motif
- C - Xlib
- D - Qt

*C. El protocolo propuesto por la Xlib.*

**2** ¿Es posible visualizar en una segunda máquina una aplicación X iniciada desde una primera?

*Sí, el software en cuestión es el cliente X. Se inicia en la primera máquina, pero utiliza los recursos del servidor X de la segunda máquina para la visualización y las interacciones.*

**3** ¿Quién define el modelo y el estilo de las ventanas?

*Es el Window Manager. Al final, X pone las bases primitivas de dibujo y gestiona los eventos, pero le corresponde a WM definir las formas y motivos que se van a aplicar.*

**4** ¿Es posible elegir el gestor de ventanas?

*Sí. Existen varios gestores de todos los estilos, según el toolkit utilizado.*

**5** ¿Qué es un Widget?

- A - Una función de gestión de ventanas.
- B - Un elemento básico de la interfaz gráfica.
- C - Un estilo de visualización.
- D - Una librería de gestión de los eventos.

*B. Es un elemento de una ventana o de la interfaz gráfica.*

**6** Cite al menos dos toolkits diferentes y habituales.

*Dos entre Xt, Motif, Qt, GTK, AWT, Athena, etc.*

**7** ¿Qué puede hacer si demasiadas ventanas están abiertas en su escritorio y no consigue aclararse?

*Utilizar los escritorios virtuales.*

**8** ¿Qué proyecto intenta federar las bases de la integración de los diferentes proyectos relacionados con el entorno de escritorio de Linux?

- A - El proyecto Xfce
- B - El proyecto KDE
- C - El proyecto GNOME
- D - El proyecto Freedesktop

*D - El proyecto Freedesktop*

## **Xorg**

**9** ¿De qué proyecto deriva X.org?

*De XFree86.*

**10** ¿Dónde se encuentra el fichero de configuración de Xorg?

*Es el fichero /etc/X11/xorg.conf.*

**11** ¿Cuál es el mejor método para instalar Xorg?

*Debería utilizar los paquetes oficiales o adicionales previstos para su distribución. La compilación de Xorg es muy larga.*

**12** ¿Cuáles son los cuatro medios disponibles para configurar X Window presentados en este libro?

*Mediante las herramientas de la distribución (p. ej.: SaX), mediante Xorgcfg, mediante xorgconfig (antiguamente xf86config) y editando el fichero de configuración a mano.*

**13** ¿Cuál es la sección del fichero de configuración de Xorg que permite configurar un teclado o un ratón?

- A - InputDevice
- B - Mouse
- C - Keyboard
- D - Modules

*A. La sección InputDevice permite configurar cualquier tipo de periférico de entrada y no se limita solamente al teclado y el ratón.*

**14** ¿Qué opción relacionada con el driver kbd permite modificar el idioma del teclado?

*Es la opción XkbLayout. Ponga su valor a «es» para un teclado español.*

**15** Su ratón dispone de una rueda constituida por la combinación de los botones 4 y 5. ¿Qué opción debe añadir en la sección correspondiente de Xorg?

*Es la opción ZaxisMapping, que toma los valores «4 5».*

**16** ¿Es obligatorio especificar las frecuencias del monitor?

*No; si es de tipo DDC y el driver lo soporta, el monitor devuelve sólo la información correcta.*

**17** Su monitor soporta una resolución que Xorg no reconoce. ¿Qué posibilidad tiene para «forzar» a Xorg a utilizar esta resolución?

*Debe crear o recuperar el modeline correspondiente a esta resolución. La lista de los sitios proporcionados en el capítulo La seguridad puede ayudarle.*

**18** La zona de visualización es demasiado ancha y los botones de ajustes de la pantalla no arreglan nada. ¿Qué herramienta puede ayudarle a rectificar la anchura de la visualización?

*Es xvidtune. Hay riesgo real de estropear el monitor.*

**19** ¿De qué dependen las opciones de la sección Device?

*Del driver de la tarjeta gráfica. Las opciones no son las mismas de un driver a otro.*

**20** ¿Cuál es el papel de la sección Screen?

- A - Describe el monitor.
- B - Describe la tarjeta gráfica.
- C - Describe los parámetros de visualización.
- D - Describe la configuración global del servidor X.

*C. La sección Screen agrupa las secciones Device (tarjeta gráfica) y Monitor (monitor), y configura tanto las resoluciones que el servidor X debe proponer como la profundidad de los colores por defecto.*

**21** Su servidor X debe arrancar en 1.600x1.200 por defecto y proponer las resoluciones 1.280x1.024 y 1.024x768. ¿Cuál es la línea asociada a «Modos»?

*Coloque la resolución por defecto en primero. Modos "1600x1200" "1280x1024" "1024x768".*

**22** ¿Cuál es el papel de la sección ServerLayout?

*Asocia el conjunto de los periféricos, visualización y entradas/salidas, para proponer una configuración de hardware coherente.*

**23** No quiere llenar las rutas FontPath del fichero de configuración X. ¿Dónde colocará sus fuentes para que sean incorporadas automáticamente?

*Coloque un vínculo hacia el directorio de sus fuentes en /etc/X11/fontpath.d.*

**24** ¿Cuáles son los tres módulos que, en teoría, deben cargarse en memoria para activar las extensiones OpenGL y el soporte de hardware 3D?

*Debería cambiar los módulos GLcore, glx y dri. No se puede garantizar que su driver de tarjeta gráfica soporte la aceleración de hardware.*

**25** ¿Cómo impedir que un usuario desconfigure su servidor X?

*Ponga la opción «DontZap» a verdadero en la sección «ServerFlags».*

**26** El inicio de X Window ha sido tan directo que devuelve un error relacionado con su parámetro -probeonly. ¿Qué tiene que hacer?

*Debe analizar el contenido del fichero /var/log/Xorg.0.log, que le indicará el motivo del fracaso en el inicio del servidor X.*

**27** El inicio de X lleva a la visualización de una pantalla gris con una cruz en medio y nada más. ¿Qué está pasando?

*Es normal. Usted debe iniciar X Window desde un Display Manager o desde el comando starx o con cualquier*

*otro script de su invención, que instalará un entorno como un gestor de ventanas o un entorno de escritorio. Solo, X no sabe hacer nada.*

### **Los gestores de visualización XDM**

**28** ¿Para qué sirve XDM?

- A - Abrir una sesión gráfica local.
- B - Abrir una sesión gráfica remota.
- C - Autenticar a un usuario.
- D - Gestionar múltiples sesiones X.

*Todas las afirmaciones, A, B, C y D, son correctas.*

**29** ¿Cuál es el nombre del protocolo X que permite gestionar las conexiones y aperturas de sesiones remotas?

*XDMCP, X Display Manager Control Protocol.*

**30** ¿Es posible iniciar KDE desde GDM?

*Sí. El entorno de trabajo del usuario es independiente del Display Manager. Incluso puede proponerle una elección entre varios entornos.*

**31** Señale algunas diferencias entre los Display Manager de KDE o GNOME y xdm.

*Los dos primeros proponen una lista de usuarios, iconos asociados, elección de las sesiones, autoconexión, temas visuales «simpáticos», etc.*

**32** Dé la ruta (directorio) de los ficheros de configuración de xdm.

*Los ficheros de configuración de xdm están en /etc/X11/xdm.*

**33** ¿Cómo se llaman los parámetros que permiten ajustar xdm?

- A - variables;
- B - argumentos;
- C - recursos;
- D - parámetros.

*C. Como para cualquier producto de X Window, se habla de los recursos X de este producto.*

**34** ¿Cuál es el contenido principal del fichero xdm-config?

*Esencialmente, las ubicaciones de los demás recursos utilizados por xdm.*

**35** ¿En qué momento se ejecuta el comando Xsetup?

*Se ejecuta el comando Xsetup antes de la aparición del cuadro de login, en particular para modificar su estilo.*

**36** ¿Cuál es el primer fichero que se ejecuta en el entorno del usuario después de su conexión?

*Se ejecuta primero el script Xsession en el espacio del usuario, después de su conexión. No confunda este script con Xstartup, que se ejecuta con root después de la autenticación de la persona, pero antes de la apertura de la sesión de usuario.*

**37** ¿Dónde puede colocar su propia configuración de sesión el usuario común?

- A - ~/xinitrc

- B - `~/.xsession`
- C - `~/.xstartup`
- D - `~/.xresources`

*A o B, teniendo en cuenta que se busca primero B .xsession y que, en caso de encontrarlo, no se ejecuta .xinitrc.*

**38** ¿Cómo asegurarse de encontrar el fichero de configuración de kdm o gdm?

*Como hay cada vez más sitios posibles según la distribución y el lugar de instalación de KDE o de GNOME, piense en utilizar el comando find o locate para buscar el fichero kdmrc o gdm.conf.*

**39** ¿Cuál es el nivel de ejecución para arrancar en modo gráfico en la mayoría de las distribuciones?

- A - 3
- B - 4
- C - 5
- D - Irrelevante: xdm se ejecuta mediante un script en /etc/init.d.

*C. Se inicia Xdm (o kdm, gdm) en el nivel de ejecución 5, por convención. La respuesta D podría ser válida, y es el caso de algunas distribuciones, y nada podría impedir iniciar este servicio en cualquier nivel. Pero por convención sólo está activo en el nivel 5.*

**40** ¿Cómo determina gdm o kdm qué tipo de sesión por defecto se debe iniciar al conectarse un usuario determinado?

- A - Por el contenido del directorio /usr/share/xsessions.
- B - Por el contenido del fichero /etc/sysconfig/desktop.
- C - Por la variable del sistema DISPLAYMANAGER.
- D - por la lectura del contenido del fichero `~/.dmrc`.

*D. El fichero `.dmrc` contiene el nombre de la última sesión abierta por el usuario, que debe ponerse en correspondencia con el fichero del mismo nombre en /usr/share/xsessions.*

**41** ¿Qué herramienta sustituye la consola en X?

*El terminal de tipo xterm o un equivalente.*

**42** ¿Cuántos gestores de ventanas hay?

- A - 4
- B - 7
- C - 15
- D - difícil de saber.

*D: es difícil saber cuántos gestores de ventanas existen. Hay decenas.*

**43** ¿Cómo exportar la visualización de todos los programas X en la primera pantalla del primer servidor X 10.17.32.142?

*Exporte la variable DISPLAY como a continuación: `export DISPLAY=10.17.32.142:0.0`*

## **Recursos X**

**44** ¿Cómo iniciar xcalc con las teclas en fondo marrón (brown)?

`xcalc -bg brown`

**45** ¿Cómo conocer la «clase» de una ventana X para luego poder modificar sus recursos?

*Primero inicie el comando/ventana X y luego inicie el comando xprop desde un terminal X. Con el cursor de ratón en cruz, haga clic en la ventana deseada. En la salida del terminal, áísle la línea WM\_CLASS. Da la clase de la aplicación.*

**46** ¿Cómo colocar en un fichero xclock.ressources el conjunto de los recursos del proceso de clase xclock (el comando xclock)?

*Ejecute el comando appres xclock > xclock.ressources.*

**47** ¿Cómo modificar de manera dinámica y gráfica los recursos de una aplicación X iniciada?

*Ejecute el comando editres y en el menú de Commands seleccione Get tree y haga clic en la ventana X correspondiente.*

**48** El recurso que permite modificar el color de la aguja de las horas en xclock es «xclock.clock.hourColor». ¿Cómo lograr que a partir de las próximas conexiones y de manera definitiva el color de la aguja sea negro?

*Añada «xclock.clock.hourColor: black» en el fichero .Xdefaults. Luego teclee (para la sesión actual) xrdb -merge .xdefaults. Se leerá automáticamente el fichero .Xdefaults a cada apertura de sesión.*

# Prácticas

## 1. Entender X Window

1. Desde un boot en modo consola (init 3, por ejemplo), inicie el servidor X Window en segundo plano. Debe ver una pantalla gris y una cruz en medio. ¿Cuáles son las acciones posibles?

```
$ X &
```

Excepto mover el ratón, nada funciona: no hay ni gestor de ventanas, ni programas iniciados.

2. Teclee [Alt][Ctrl][F1]. Ahora está de nuevo en la consola en modo texto. Exporte la variable DISPLAY, que permite indicar a los clientes X las «coordenadas» del servidor X.

```
$ export DISPLAY=localhost:0
```

3. Todavía desde la consola, después de haber exportado la variable, ejecute el programa twm en segundo plano:

```
$ twm &
```

4. Use en X Window la combinación [Alt][F7]. Haga clic en el botón izquierdo del ratón. ¿Qué está pasando?

Ha iniciado twm, un gestor de ventanas básico. El botón derecho le permite acceder a un menú. En este menú, haga clic en Xterm o equivalente.

La ventana xterm se dibuja. Haga clic en la pantalla para ubicarla.

5. Desde xterm, teclee el comando xclock.

```
$ xclock &
```

Ubique la nueva ventana. Para funcionar, X necesita un entorno básico, el gestor de ventanas. Este gestor se apoya en los widgets para el dibujo de las ventanas, y X ofrece las funciones primitivas de dibujo y control de eventos gráficos.

## 2. Configuración simple de Xorg

Objetivo: adaptar X.org a sus necesidades modificando /etc/X11/xorg.conf. X debe estar ya instalado y debe poder iniciarse.

1. En el momento de seleccionar el español como esquema de teclado para X, a menudo se configura éste como un teclado de 102 teclas, haciendo que las teclas «Windows» no estén disponibles. Estas teclas se llaman teclas «Meta» en X. Modifique la configuración de X Window para un teclado de 105 teclas (el contenido de su sección puede variar) en la sección InputDevice que contiene el driver kbd:  
Option "XkbModel" "pc105"

2. Cuando presiona Caps Lock, el funcionamiento difiere en X, en comparación con Windows; accede, por ejemplo, a las mayúsculas acentuadas: Á, Í, Ó, Ú, y así sucesivamente. Puede modificarlo en la misma sección indicando que CapsLock funcione como un Shift bloqueado:  
Option "XkbOptions" "caps:shiftlock"  
Luego vuelva a ejecutar X.

3. Su ratón dispone de 9 botones. En X, inicie el programa xev en un terminal (xterm, konsole, etc.):

```
$ xev
```

En una nueva ventana vacía, se visualiza un rectángulo con bordes negros. Mueva el cursor del ratón dentro y haga clic en todos los botones. Observe el texto que está desfilando: ¿se reconocen todos sus botones?

Si no es el caso, modifique la sección InputDevice correspondiente a su ratón:

```
Option "Buttons" "5"
```

Reinic peace de nuevo.

4. Se configura a menudo un monitor LCD (plano) con una resolución por defecto, por ejemplo 1.280x1.024 para una 19 pulgadas 4/3. Intente cambiar de resolución con las combinaciones de teclas [Ctrl][Alt] + y [Ctrl][Alt] - . Si nada se mueve, debe modificar la subsección Display correspondiente a su monitor.

Coja el manual del monitor para conocer las resoluciones soportadas. En la sección Screen, mire el valor de DefaultDepth, y apunte la sección Display, que contiene el mismo valor Depth. Modifique las resoluciones según el manual de su monitor:

Modes "1280x1024" "1024x768" "800x600" "640x480"

Reinic peace de nuevo.

5. Es posible que la visualización de sus fuentes esté deformada. X no conoce obligatoriamente el tamaño real (en cm) de su pantalla y el número de DPI no es correcto. Modifique la sección Monitor modificando o añadiendo la línea siguiente:

DisplaySize xxx yyy

xxx e yyy son la anchura y la longitud de visualización, en milímetros. Vuelva a iniciar X y mire si se ha modificado el tamaño de las fuentes.

### 3. Configuración simple del Display Manager

Objetivo: entender y modificar de manera básica la configuración de xdm, gdm o kdm. X debe funcionar correctamente.

1. Va a modificar la pantalla de inicio del cuadro de diálogo xdm. Busque primero el fichero que contiene los recursos de la visualización de xdm. En el fichero /etc/X11/xdm/xdm-setup, busque la línea correspondiente al parámetro siguiente:

DisplayManager\*resources: /etc/X11/xdm/Xresources

Edite el fichero apuntado, aquí /etc/X11/xdm/Xresources, aunque puede ser diferente en su caso.

2. En el fichero Xresources o equivalente, busque la línea que comienza por xlogin\*greeting: xlogin\*greeting: Bienvenido/a a CLIENTHOST

Modifique el texto. Se cambiará el mensaje de bienvenida en consonancia. CLIENTHOST es el nombre de la máquina del cliente.

3. El fichero Xsession (o equivalente) inicia la sesión X del usuario. Para ello, busca el fichero ~/.xsessions o ~/.xinitrc en la carpeta del usuario. KDM y GDM buscan el contenido del fichero ~/.dmrc. Visualice el contenido de este fichero. Por ejemplo:

```
[Desktop]
Session=kde
```

4. Muévase a /usr/share/xsessions y liste su contenido. Encuentre el fichero xxx.desktop, donde xxx equivale al valor Sesión del fichero .dmrc, y visualice su contenido, por ejemplo kde.desktop:

```
[Desktop Entry]
Encoding=UTF-8
Type=XSession
Exec=/opt/kde3/bin/startkde
TryExec=/opt/kde3/bin/startkde
Name=KDE3
...
(voluntariamente truncado)
```

5. Cuando selecciona un tipo de sesión en GDM o KDM, son estos ficheros los que se leen y cuyos nombres (Name) se visualizan. Cuando valida la elección de una sesión, se ejecuta el comando asociado al valor de Exec.

## 4. Modificar los recursos de una aplicación X

Objetivo: modificar la visualización de aplicaciones X o afiliadas. Si es posible, KDE debería funcionar.

1. Las aplicaciones X «puras» no son las únicas que soportan la modificación de su estilo. En KDE es posible modificar el estilo de los programas y los colores de cada componente mediante el centro de configuración. Abra el programa llamado KColorChooser:  
\$ kcolorchooser &  
Inicie también la calculadora de KDE:  
\$ kcalc &
2. Desde el selector de los colores, elija un color, luego arrastre y suelte este color en una tecla de la calculadora. ¿Qué está pasando?  
La tecla y las que son de la misma clase cambian de color. Adoptan el color seleccionado.
3. Los recursos de las aplicaciones KDE no se apoyan directamente en los recursos X, sino en toolkits, entre otros Qt. En el menú configuración de kcalc, seleccione «colores» y, como en el punto 3, establezca los colores en los cuadros de los colores de los botones; luego haga clic en **OK** o **Aplicar** y se guardarán sus ajustes.

# Requisitos y objetivos

## 1. Requisitos

- Disponer de un acceso con privilegios de root.
- Tener uno o varios discos inutilizados o, en su defecto, una zona libre no particionada en un disco.

## 2. Objetivos

Al final de este capítulo, usted será capaz de:

- Entender las diferencias entre los niveles de RAID.
- Crear y modificar matrices RAID.
- Gestionar los fallos y modos deteriorados.
- Instalar un LVM.
- Modificar a voluntad el tamaño de sus volúmenes lógicos.
- Ampliar y reducir sus sistemas de ficheros.

# Particionamiento avanzado RAID

## 1. Definiciones

El **RAID** (*Redundant Array of Inexpensive Disks*) ha sido definido por la Universidad de Berkeley en 1987 con el doble objetivo de reducir los costes y aumentar la fiabilidad del almacenamiento de datos. El objetivo es compaginar varios pequeños discos físicos independientes en una matriz (array: tabla, conjunto, fila, matriz) de discos cuya capacidad supera la del SLED (*Single Large Expensive Drive*). Una matriz aparece como una unidad lógica de almacenamiento único.

El **MTBF** (*Mean Time Between Failure* - intervalo entre fallos) del conjunto es igual al MTBF de un disco individual dividido por el número de discos en el conjunto, y por lo tanto, teóricamente, una solución RAID puede no ser la adecuada en sistemas críticos. Afortunadamente, un sistema RAID es tolerante a los fallos gracias a que almacena de manera redundante su información según varios métodos:

- **RAID-0**, llamado **stripe mode**: al menos dos discos forman un único volumen. En principio, los dos discos tienen el mismo tamaño. Se fraccionará y efectuará cada operación de lectura/escritura en cada uno de los discos. Por ejemplo, se escribirán 4 KB en el disco 0, 4 KB en el disco 1, 4 KB en el disco 2, luego 4 KB en el disco 0, etc. Así, se aumentan las prestaciones, ya que se efectúan en paralelo las operaciones de lectura y escritura sobre los discos. Si N es el número de discos y P la velocidad de transferencia, la velocidad de transferencia del volumen RAID es, en principio, próxima a  $N \times P$  mbps. El RAID-0 no tiene ninguna redundancia. En caso de fallo de uno de los discos, es probable que se pierda el conjunto de los datos.
- **RAID-1**, llamado **mirroring**: primer modo redundante. Se puede utilizar a partir de dos discos o más con posibles discos de emergencia (*Spare Disk*). Se duplica cada información escrita en un disco sobre los demás. Si N-1 discos del RAID caen, los datos siguen intactos. Si hay un disco de emergencia, en caso de fallo, se reconstruye automáticamente y se sustituye el disco que falla. Las prestaciones en escritura pueden ser malas: escritura en N discos al mismo tiempo, con el riesgo de saturar el controlador de disco y el bus. Las prestaciones en lectura son buenas, porque RAID emplea un algoritmo que puede leer los datos en cada disco (ya que son idénticos).
- **RAID-5**: RAID con algoritmo distribuido de paridad. Es el modo más utilizado, ya que ofrece la mejor relación entre el número de discos, el espacio disponible y la redundancia. Hacen falta al menos tres discos con posibles discos de emergencia. Hay paridad en cada uno de los discos. El tamaño final es el de N-1 discos. El RAID-5 sobrevive a un fallo de disco. En este caso, si hay un disco de emergencia, será reconstruido automáticamente. Las prestaciones en lectura son equivalentes a las del RAID-0, mientras que en escritura dependen del algoritmo empleado, así como de la memoria de la máquina.

## 2. Precauciones y consideraciones de uso

### a. Disco de emergencia

Un disco de emergencia (*Spare Disk*) no forma parte integrante de una matriz RAID mientras no se averíe un disco. Si eso sucede, se marca el disco como defectuoso y el primer disco Spare toma el relevo. Pase lo que pase, conviene cambiar el disco averiado lo antes posible y volver a construir el RAID.

### b. Disco averiado

Un disco averiado (*Faulty Disk*) es un disco que ha sido reconocido como defectuoso o como erróneo por el RAID. En este caso, RAID utiliza el primer disco SPARE para reconstruir su matriz. Los discos faulty pertenecen siempre a la matriz, pero están desactivados.

## c. Boot

No se debe ubicar en una matriz RAID la partición boot (la que contiene el núcleo, la configuración del bootloader, los ficheros imágenes de discos): el cargador del sistema operativo es incapaz de montar particiones RAID (la próxima versión del GRUB sí que podrá).

## d. Swap

Es posible instalar una partición de intercambio (swap) sobre un RAID, pero no suele ser útil en los casos comunes. En efecto, Linux es capaz de equilibrar el uso del swap sobre varios discos/particiones individuales. En este caso, declare n swaps en `/etc/fstab` con la misma prioridad.

```
/dev/sda2      swap      swap    defaults,pri=1  0 0  
/dev/sdb2      swap      swap    defaults,pri=1  0 0  
/dev/sdc2      swap      swap    defaults,pri=1  0 0
```

Sin embargo, en caso de necesidad de alta disponibilidad, es posible la swap en el RAID.

## e. Periféricos

El sistema reconoce una matriz RAID como un periférico de tipo bloque, al igual que cualquier disco físico. Así, un RAID puede estar constituido por discos, particiones (en general, se crea una única partición en cada disco). El bus no tiene importancia alguna: usted puede construir una matriz RAID con discos SCSI e IDE mezclados. Asimismo, se puede construir un RAID sobre otras matrices RAID, por ejemplo RAID-0+1 (2x2 discos sobre RAID-1, formando sobre RAID-0 una nueva matriz las dos resultantes). Los periféricos RAID tienen la forma:

```
/dev/md0  
/dev/md1
```

## f. IDE

Si los discos IDE han sido mucho tiempo el SCSI del pobre (hardware de menor calidad, lentitud, falta de fiabilidad), hoy por hoy esto ya no es cierto. Los últimos modelos son totalmente idénticos a los discos SCSI, si exceptuamos el controlador. De hecho, usted puede montar configuraciones RAID en IDE a un coste razonable. Sin embargo, hay que recordar una regla:

### UN SOLO DISCO IDE POR BUS IDE

En la práctica esto corresponde a poner un disco por bus, nada más. La razón de ello estriba en que un bus IDE sobrevive al fallo en un disco, pero si lo que falla es el bus mismo se pierden todos los discos conectados al bus y con ellos la matriz RAID. La compra de tarjetas IDE adicionales (a bajo precio) permite compensar el problema de fiabilidad (dos discos por tarjeta).

El principio es el mismo que para los discos SATA. En teoría es más simple, ya que sólo puede conectar un disco por cable. Sin embargo, si lee el manual de su placa base o el del controlador, le recordarán que a menudo los puertos SATA van por parejas. Además, si su placa contiene varios controladores, puede ser una buena idea separar los discos: uno por controlador, o como con IDE, añadir un controlador SATA adicional por PCI o PCI-Express.

## g. Hot Swap

- **IDE:** ¡NUNCA DESENCHUFE UN DISCO IDE EN CALIENTE! Es la mejor manera de destruir el disco, si no fuese ya el caso, y destruir el controlador IDE (y, en ocasiones, la placa base o la IDE adicional). El disco IDE no está diseñado para ello.

- **SCSI**: los controladores SCSI no están previstos para el Hot Swap, pero en teoría deberían funcionar de todas maneras, si el disco es idéntico física y lógicamente.
- **SATA**: Se equipara el SATA con un SCSI. La especificación SATA en versión 2 soporta teóricamente el Hot Swap. Sin embargo, la mayoría de los controladores actuales o no implementan o implementan mal esta posibilidad; de ahí que haya riesgos de bug o de quemar su controlador. Compruébelo en la documentación del fabricante de su placa base (chipset).
- **SCA**: son discos SCSI específicos. Consulte el documento «Software RAID Howto».

## 3. RAID con mdadm

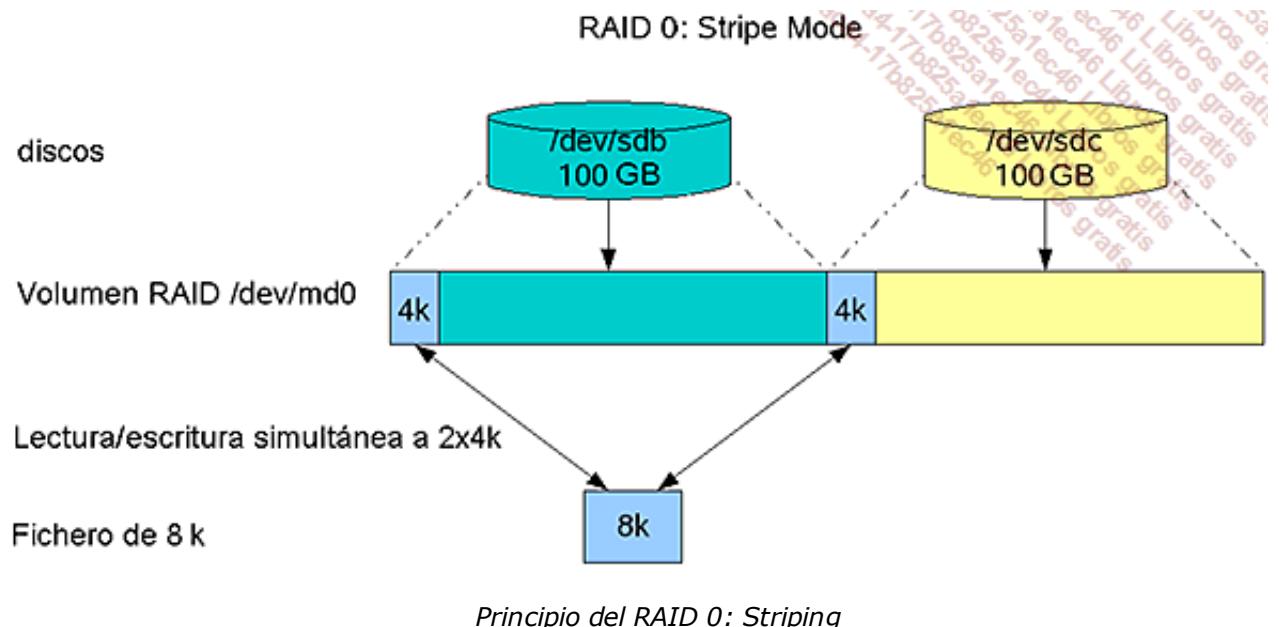
### a. Preparación

La herramienta **mdadm** sustituye a las herramientas raidtools de las antiguas distribuciones Linux. Esta herramienta es más sencilla y permite efectuar todas las operaciones. Su fichero de configuración es `/etc/mdadm.conf`.

Con el fin de crear matrices RAID, es necesario que las particiones que servirán para crear la matriz sean de tipo **0xFD** (Linux RAID autodetect). Las particiones deben estar físicamente en discos diferentes, pero para hacer pruebas el soporte RAID autoriza particiones en un mismo disco. En este caso, usted deberá tener en cuenta que las particiones dispongan del mismo tamaño.

### b. Creación

#### RAID-0



O sea, dos particiones `/dev/sdb1` y `/dev/sdc1`. Va a crear una partición RAID-0, ensamblaje de estas dos particiones.

```
# mdadm --create /dev/md0 --level=raid0 --raid-devices=2 /dev/sdb1 /dev/sdc1
--create
```

Crea un RAID.

/dev/md0

Nombre del fichero de tipo bloque que representa la matriz RAID.

--level

Tipo de RAID que se va a crear: 0, raid0 y stripe para RAID0.

➤ linear no es RAID0 (llenado poco a poco).

--raid-devices

Número de particiones utilizadas para crear la matriz.

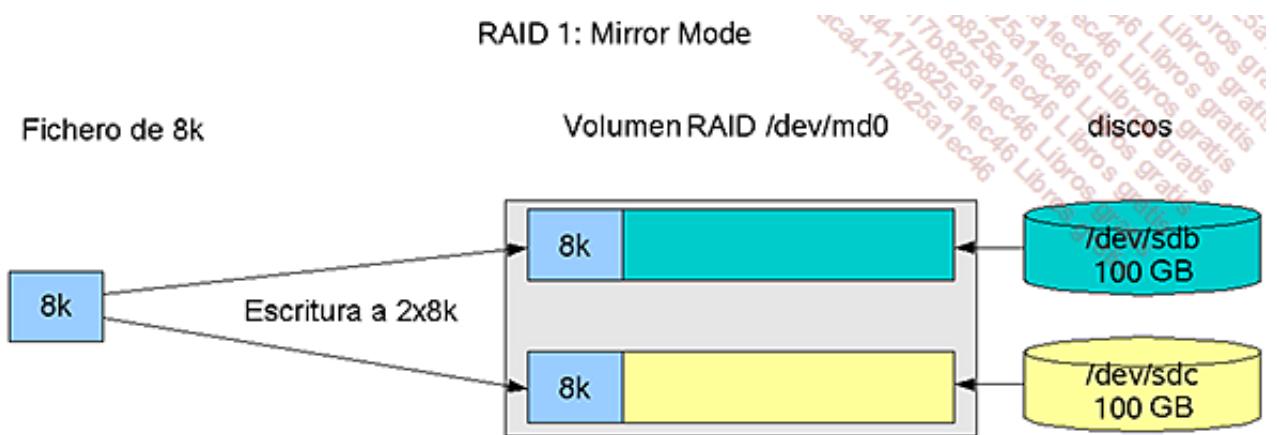
/dev/sdb1, /dev/sdc1

Particiones que forman la matriz, según el número indicado en --raid-devices.

Sólo queda instalar el sistema de ficheros en el disco RAID:

```
# mkfs -t ext4 /dev/md0
```

## **RAID-1**



El principio es el mismo. Esta vez, tendrá que añadir una partición de emergencia /dev/sdd1.

```
# mdadm --create /dev/md0 --level=raid0 --raid-devices=2 /dev/sdb1  
/dev/sdc1 --spare-devices=1 /dev/sdd1
```

--level 1, mirror o raid1 son valores aceptados para un RAID-1.

--spare-devices número de discos de emergencia para utilizar.

/dev/sdd1 partición o particiones de emergencia, según el número indicado en -spare-devices.

Luego:

```
# mkfs -t ext4 /dev/md0
```

## **RAID-0+1**

Hacen falta al menos cuatro particiones. Debe crear dos matrices RAID-1, que va a agrupar en una sola

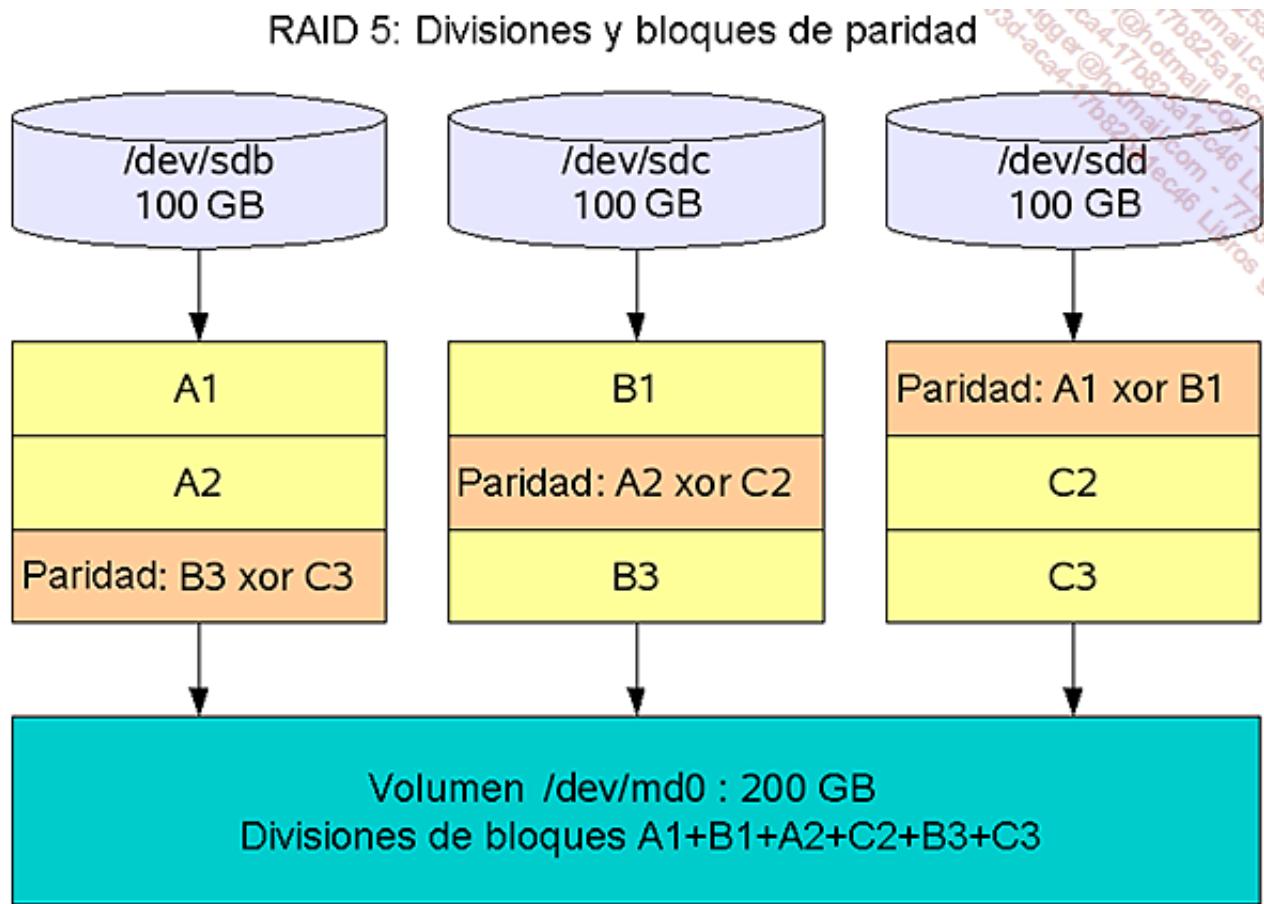
matriz RAID-0.

```
# mdadm --create /dev/md0 --level=raid1 --raid-devices=2 /dev/sdb1 /dev/sdc1  
# mdadm --create /dev/md1 --level=raid1 --raid-devices=2 /dev/sdd1 /dev/sde1  
# mdadm --create /dev/md2 --level=raid0 --raid-devices=2 /dev/md0 /dev/md1
```

Luego:

```
# mkfs -t ext4 /dev/md2
```

## **RAIDS**



El RAID va a emplear tres discos de datos /dev/sdb1, /dev/sdc1, /dev/sdd1 y un disco de emergencia /dev/sde1.

```
# mdadm --create /dev/md0 --level=raid5 --raid-devices=3 /dev/sdb1 /dev/sdc1 /dev/sdd1  
--spare-devices=1 /dev/sde1
```

Luego se instala el sistema de ficheros:

```
# mkfs -t ext4 /dev/md2
```

### **c. Guardar la configuración**

Para facilitar la tarea de la herramienta **mdadm**, puede crear (no es obligatorio) el fichero de

configuración `/etc/mdadm.conf`. Se puede crear este fichero manualmente, pero la herramienta **mdadm** sabe generarlo. Es preferible hacerlo DESPUÉS de la creación de las matrices RAID.

```
# echo "DEVICE partitions" > /etc/mdadm.conf  
# mdadm --detail --scan>> /etc/mdadm.conf
```

## 4. Estado del RAID

El fichero virtual `/proc/mdstat` contiene información sobre el RAID. En este fichero puede ver el detalle de un RAID, en particular si alguno de los volúmenes de la matriz está averiado (Faulty).

```
Personalities: [raid1]  
md0: active raid1 hda10[2] hda9[1] hda8[0]  
      104320 blocks [2/2] [UU]
```

El comando **watch** permite comprobar un estado en tiempo real:

```
# watch cat /proc/mdstat
```

También puede utilizar **mdadm** con el parámetro `--detail`:

```
# mdadm --detail /dev/md0  
/dev/md0:  
    Versión: 00.90.01  
Creation Time: Mon Jan 23 22:10:20 2006  
    Raid Level: raid1  
    Array Size: 104320 (101.88 MiB 106.82 MB)  
    Device Size: 104320 (101.88 MiB 106.82 MB)  
    Raid Devices: 2  
    Total Devices: 3  
Preferred Minor: 1  
    Persistence: Superblock is persistent  
  
    Update Time: Mon Jan 23 22:13:06 2006  
    State: clean  
Active Devices: 2  
Working Devices: 3  
Failed Devices: 0  
Spare Devices: 1  
  
    Number  Major  Minor  RaidDevice State  
        0      3        8          0     active sync   /dev/hda8  
        1      3        9          1     active sync   /dev/hda9  
        2      3       10         -1     spare          /dev/hda10  
    UUID: 90e838b5:936f18c7:39f665d3:d9dad1a9  
Events: 0.4
```

Cabe señalar que, con este último comando, puede obtener muchos más detalles, en particular cuáles son los discos «spare» y «faulty».

## 5. Simular una avería

Ahora va a simular una avería en `/dev/hda8`:

```
# mdadm /dev/md0 -f /dev/hda8
```

```
mdadm: set /dev/hda8 faulty in /dev/md0
```

Mire el estado del RAID en `/proc/mdstat` durante la ejecución:

```
md0: active raid1 hda10[2] hda9[1] hda8[0](F)
      104320 blocks [2/1] [U_]
      [=.....] recovery = 8.8% (9216/104320) finish=0.1min
speed=9216K/sec
```

Ha aparecido una «(F)» cerca de `hda8`, lo que indica un disco Faulty. Vemos también que, de los dos discos, uno tiene una avería y que el RAID reconstruye su matriz con el spare disk. Después de la ejecución, obtenemos:

```
md0: active raid1 hda10[1] hda9[0] hda8[2](F)
      104320 blocks [2/2] [UU]
```

RAID está reconstruido y funciona de maravilla.

```
# mdadm --detail /dev/md0
...
      State: clean
 Active Devices: 2
Working Devices: 2
 Failed Devices: 1
  Spare Devices: 0

      Number  Major  Minor  RaidDevice State
          0      3        9        0     active sync   /dev/hda9
          1      3        10       1     active sync   /dev/hda10
          2      3        8       -1    faulty       /dev/hda8
...
...
```

disco Faulty es realmente `/dev/hda8`; `/dev/hda10` lo sustituyó como disco de emergencia. Así, el disco de emergencia se convierte en un disco RAID de la matriz.

## 6. Sustituir un disco

Ya que `/dev/hda8` tiene un problema, lo va a sustituir. Sáquelo con `-r` (o `--remove`):

```
# mdadm /dev/md0 -r /dev/hda8
mdadm: hot removed /dev/hda8

# cat /proc/mdstat
Personalities: [raid1]
md0: active raid1 hda10[1] hda9[0]
      104320 blocks [2/2] [UU]
```

Constate que `hda8` ha desaparecido. Puede apagar la máquina y luego sustituir el disco defectuoso. Encienda de nuevo la máquina, y vuelva a particionar el disco correctamente. Sólo falta añadir de nuevo el disco arreglado en la matriz RAID con `-a` (`--add`):

```
# mdadm /dev/md0 -a /dev/hda8
mdadm: hot added /dev/hda8

# cat /proc/mdstat
Personalities: [raid1]
```

```
md0: active raid1 hda8[2] hda10[1] hda9[0]
      104320 blocks [2/2] [UU]
```

disco `hda8` aparece de nuevo. Vea el detalle:

```
# mdadm --detail /dev/md0
...
      State: clean
 Active Devices: 2
Working Devices: 3
 Failed Devices: 0
 Spare Devices: 1

      Number  Major  Minor  RaidDevice State
          0      3        9        0     active sync  /dev/hda9
          1      3       10        1     active sync  /dev/hda10
          2      3        8       -1     spare           /dev/hda8
...
```

Se ha añadido de nuevo el disco `/dev/hda8` y el sistema lo ha convertido en el nuevo disco de emergencia!

## 7. Apagado y puesta en marcha manual

Puede apagar de manera puntual una matriz RAID con `-s (--stop)` DESPUÉS de haber desmontado el periférico:

```
# mdadm --stop /dev/md0
```

La matriz RAID se pone en marcha de nuevo con `-As (--assemble -scan)`. Eso implica que el fichero `/etc/mdadm.conf` está correctamente configurado (`--scan` es una opción que busca la información en ese fichero).

```
# mdadm --assemble --scan /dev/md0
```

Si el RAID no arranca de nuevo, puede intentar con `-R (--run)`: es probable que falte un disco o que no se haya acabado aún una reconstrucción en curso:

```
# mdadm --run /dev/md0
```

# Iniciación al LVM

## 1. Principio

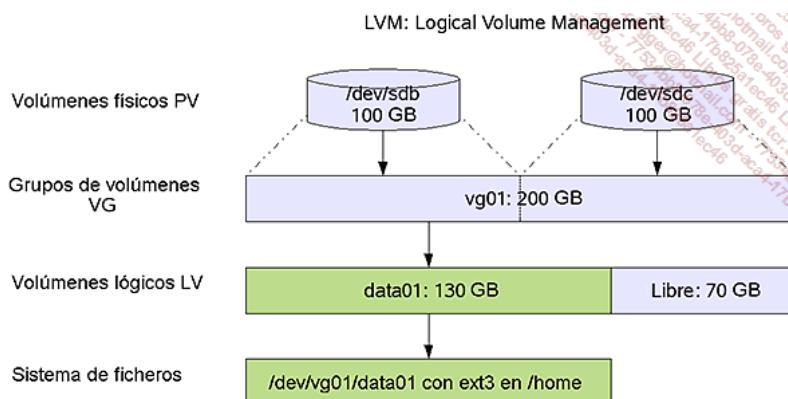
El **Logical Volume Manager** es un sistema de gestión muy perfeccionado de los soportes de almacenamiento. El objetivo es superar, incluso trascender, la gestión física de los discos y su organización lógica básica (las particiones) para extender la capacidad global de los soportes, usando una gestión completamente lógica de ella.

Un LVM permite, del mismo modo que el RAID 0 por ejemplo, crear espacios de datos lógicos en varios discos. Permite también hacer mirroring, como el RAID 1. Pero la comparación no va más allá. El RAID se limita a crear una «partición» en un espacio de almacenamiento definido por el propio RAID (por ejemplo, una partición de 100 GB en un RAID 0 de dos discos de 50 GB).

El LVM agrupa discos físicos o cualquier otro soporte de almacenamiento denominado físico (disco, hardware RAID, software RAID, soporte de almacenamiento procedente de un SAN), que llama a volúmenes físicos PV (*Physical Volume*) en un grupo de volúmenes VG (*Volume Group*). El LVM ve el grupo VG como una especie de metadisco en el cual usted va a crear volúmenes lógicos LV (*Logical Volume*) a voluntad.

- Volumen físico PV: un soporte físico de almacenamiento de datos: disco duro, por ejemplo;
- Grupo de volúmenes VG: una agrupación lógica de 1 a n VG;
- Volumen lógico LV: un recorte lógico en el interior de un VG.

Se ve el volumen lógico como una partición y se puede utilizar como tal. Puede contener datos; basta con crear un sistema de ficheros ordinario (ext3, por ejemplo) y montarlo de manera totalmente clásica.



Al contrario del Software RAID 0, donde la partición de datos debe ocupar todo el espacio, es posible crear tantos volúmenes lógicos de cualquier tamaño como se desee. Pero va mucho más allá.

El LVM es dinámico. Es posible añadir y suprimir volúmenes físicos de un grupo de volúmenes. Al añadir volúmenes físicos, la capacidad y por lo tanto el espacio disponible del grupo aumenta. El nuevo espacio disponible puede permitir crear nuevos volúmenes lógicos, pero también ampliar un volumen lógico existente.

Un volumen lógico es dinámico: se puede ampliar o reducir a voluntad, lo que implica que también debe poderse ampliar un sistema de ficheros o reducirlo.

Apunte también que se puede utilizar una matriz RAID como volumen físico.

Las configuraciones de LVM se encuentran repartidas por los ficheros y directorios de `/etc/lvm`. El fichero `/etc/lvm/lvm.conf` contiene la configuración global. La configuración de los diferentes volúmenes (físicos, grupos y lógicos) no se encuentra en un fichero, sino en una estructura presente en el interior de los propios periféricos, en sus primeros bloques: son los metadatos de los volúmenes físicos.

## 2. Los volúmenes físicos

### a. Crear un volumen físico

Un volumen físico puede ser un disco completo o una partición clásica en el interior de un disco. En este caso, la partición debe ser de tipo **0x8e**.

A continuación tiene la salida del comando **fdisk** en `/dev/sdb`. Haga la distinción entre las particiones primarias 2 y 3 de tipo **8e** que servirán para los ejemplos siguientes.

```
# fdisk -l /dev/sdb
Disk /dev/sdb: 160.0 GB, 160041885696 bytes
255 heads, 63 sectors/track, 19457 cylinders
Units = cilindros de 16065 * 512 = 8225280 bytes
Disk identifier: 0x000eab03

Períferico Inicio Principio Fin Bloques Id Sistema
/dev/sdb1 2 16846 135307462+ 83 Linux
/dev/sdb2 16847 18152 10490445 8e Linux LVM
/dev/sdb3 18153 19457 10482412+ 8e Linux LVM
```

Una vez creadas las particiones, utilice el comando **pvcreate** en una primera partición (se pueden precisar varias particiones):

```
# pvcreate /dev/sdb2
Physical volume "/dev/sdb2" successfully created
```

### b. Ver los volúmenes físicos

El comando **pvdisplay** permite visualizar el conjunto de los volúmenes físicos accesibles en su sistema. También puede llevar el nombre de volumen específico.

```
# pvdisplay /dev/sdb2
"/dev/sdb2" is a new physical volume of "10,00 GB"
-- NEW Physical volume --
PV Name          /dev/sdb2
VG Name
```

De momento, la información es reducida. El PV no pertenece todavía a ningún grupo de volúmenes (línea **VG Name**). Su tamaño es de 10 GB. Las líneas más interesantes, aunque de momento vacías porque el PV no pertenece a ningún grupo de volúmenes, son las aquellas donde está indicado **PE**. PE significa Physical Extend, extensión física. Cada VG, y por lo tanto PV que lo constituye, está cortado en secciones llamadas

PV Size	10,00 GB
Allocatable	NO
PE Size (KByte)	0
Total PE	0
Free PE	0
Allocated PE	0
PV UUID	KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra

PE. El PE es la unidad básica de trabajo del LVM. Si un PE es de 4 MB, esto significa que se podrá distribuir el espacio en el interior del grupo de volúmenes por secciones de 4 MB. La asignación se hace por PE: por lo tanto, la creación de un volumen lógico de 500 PE de 4 MB da 2.000 MB.

Estos valores pasan a cero en cuanto el PE sea integrado en un VG.

### 3. Los grupos de volúmenes

#### a. Crear un grupo de volúmenes

Para crear un grupo de volúmenes, debe disponer de al menos un volumen físico. Puede crear un grupo de volúmenes con el comando **vgcreate**. Un grupo de volúmenes llevará el nombre que usted desee. Es el primer argumento del comando. Luego facilite como argumento la lista de los volúmenes físicos que componen el grupo de volúmenes, aquí sólo uno, **/dev/sdb2**.

```
# vgcreate vg01 /dev/sdb2
Volume group "vg01" successfully created
```

#### b. Propiedades de un VG

El grupo de volúmenes dispone de numerosas propiedades. Puede listarlas con el comando **vgdisplay**.

```
# vgdisplay vg01
--- Volume group ---
VG Name           vg01
System ID         lvm2
Format            lvm2
Metadata Areas    1
Metadata Sequence No 1
VG Access          read/write
VG Status          resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size           10,00 GB
PE Size            4,00 MB
Total PE          2561
Alloc PE / Size   0 / 0
Free PE / Size    2561 / 10,00 GB
VG UUID           dZt8KP-xwol-5mb3-NaVW-Wsui-3sQy-p8kvpG
```

Fíjese en las líneas MAX LV y MAX PV. La primera indica el número máximo de volúmenes lógicos que se podrá crear en este grupo de volúmenes. El valor cero indica un número infinito en teoría. La segunda indica el número máximo de volúmenes físicos que se pueden añadir al grupo de volúmenes. Aquí también el cero indica un número infinito.

Usted debe fijarse en el hecho de que Linux representa un caso particular en este campo. El número de LV y PV es aquí virtualmente infinito (como, por ejemplo, el volumen lógico lvm2), lo que no es el caso en absoluto de los demás UNIX, donde los valores MAX LV, MAX PV y PE Size se determinan automáticamente en el momento de la creación de volúmenes, y en función, entre otras, de las propiedades de los volúmenes físicos que componen el grupo. Se puede establecer los valores al crear el grupo de volúmenes usando los siguientes parámetros del comando **pvcreate**:

-I Número máximo de volúmenes lógicos

-p Número máximo de volúmenes físicos

-s Tamaño de las extensiones físicas (con un sufijo k, m, g o t para precisar la unidad).

Las últimas líneas se refieren a las PE (extensiones físicas). El grupo dispone actualmente de 2.561 extensiones de 4 MB, o sea 10 GB, todas libres. Los volúmenes lógicos ocuparán más tarde un determinado número de estas PE, según su tamaño.

El comando **vgdisplay** acepta el parámetro **-v**, que da más detalles, y en particular la lista de los volúmenes físicos que lo componen.

```
# vgdisplay -v vg01
Using volume group(s) on command line
  Finding volume group "vg01"
--- Volume group ---
VG Name           vg01
System ID         lvm2
Format            lvm2
Metadata Areas    1
Metadata Sequence No 1
VG Access          read/write
VG Status          resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size           10,00 GB
PE Size            4,00 MB
Total PE          2561
Alloc PE / Size   0 / 0
Free PE / Size    2561 / 10,00 GB
VG UUID           dZt8KP-xwol-5mb3-NaVW-Wsui-3sQy-p8kvpG

--- Physical volumes ---
PV Name           /dev/sdb2
PV UUID           KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra
PV Status          allocatable
Total PE / Free PE 2561 / 2561
```

Ahora que el PV **/dev/sdb2** forma parte de un VG, hay más información disponible:

```
# pvdisplay /dev/sdb2
--- Physical volume ---
PV Name           /dev/sdb2
VG Name           vg01
PV Size           10,00 GB / not usable 589,00 KB
Allocatable       yes
PE Size (KByte)  4096
Total PE          2561
Free PE           2561
Allocated PE      0
PV UUID           KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra
```

### 4. Los volúmenes lógicos

## a. Crear un volumen lógico

Un volumen lógico es una sección de un VG (grupo de volúmenes) o, lo que es lo mismo, una partición en la cual usted podrá crear un sistema de ficheros. Un volumen lógico LV ocupa un determinado número de PE (extensiones físicas) de un VG, contiguas o no. Esto tiene su importancia porque:

- es posible ampliar un LV mientras quedan PE libres en el VG.
- es posible reducir un LV, lo que liberará unas PE en el VG, que podrán ser utilizadas para crear nuevos LV o para ampliarlos.

Esto significa que el LVM gestiona una especie de índice y orden de PE para determinar a qué LV pertenece un PE.

Puede crear un volumen lógico con el comando **lvcreate**. Un volumen lógico lleva un nombre, dispone de un tamaño expresado en extensiones lógicas LE (*Logical Extension*) que corresponden a PE en el interior de un LV, sea en KB, MB, GB... El comando siguiente crea un volumen lógico llamado **data01** en el interior del VG **vg01**, de un tamaño de 6 GB. La **-L** precisa que la unidad está en MB (**m**), GB (**g**), TB (Terabyte, **t**), PB (Petabyte) o EB (Exabyte). Para precisar un número de PE, utilice **--l**.

```
# lvcreate -n data01 -L 6g vg01
Logical volume "data01" created
```

El sistema muestra un LV como una partición y después de su creación dispone de un fichero periférico asociado. El fichero está en la carpeta **/dev/<nombre\_del\_vg>/<nombre\_del\_lv>**. Observe que se trata de un vínculo simbólico hacia un fichero de **/dev/mapper** para guardar una compatibilidad con los demás Unix.

```
# ls -l /dev/vg01/data01
lrwxrwxrwx 1 root root 23 sept. 13 09:27 /dev/vg01/data01 -> /dev/
mapper/vg01-data01
```

## b. Propiedades de un volumen lógico

Las propiedades de un volumen lógico están accesibles por el comando **lvdisplay**:

```
# lvdisplay /dev/vg01/data01
File descriptor 3 left open
File descriptor 4 left open
--- Logical volume ---
LV Name           /dev/vg01/data01
VG Name           vg01
LV UUID           6ucPwc-sxMJ-K9P3-MkWR-t28I-NyRM-ZKKTmm
LV Write Access   read/write
LV Status         available
# open            0
LV Size           6,00 GB
Current LE        1536
Segments          1
Allocation        inherit
Read ahead sectors auto
- currently set to 256
Block device      253:0
```

Puede añadir los parámetros **-v** y **-m**. En este último caso, **lvdisplay** muestra también los segmentos ocupados por el volumen lógico en el interior de los diferentes volúmenes físicos y, por lo tanto, el reparto de las extensiones físicas ocupadas por el volumen lógico en el interior de cada volumen físico. Como, de momento, sólo hay un PV en el interior del VG, puede obtener lo siguiente:

```
# lvdisplay -m /dev/vg01/data01
...
--- Segments ---
Logical extent 0 to 1535:
  Type            linear
  Physical volume /dev/sdb2
  Physical extents 0 to 1535
```

## c. Acceso al volumen lógico

Puede crear un sistema de ficheros y montar el LV como cualquier partición:

```
# mkfs -t ext3 /dev/vg01/data01
mke2fs 1.40.8 (13-Mar-2008)
Warning: 256-byte inodes not usable on older systems
Etiqueta de sistema de ficheros=
Tipo de sistema operativo: Linux
Tamaño de bloque=4096 (log=2)
Tamaño de fragmento=4096 (log=2)
393216 i-nodos, 1572864 bloques
78643 bloques (5.00%) reservados para el superusuario
```

Sólo queda montar el nuevo sistema de ficheros.

```
Primer bloque de datos=0
Número máximo de bloques del sistema de ficheros=1610612736
48 grupos de bloques
32768 bloques por grupo, 32768 fragmentos por grupo
8192 i-nodos por grupo
Superbloques de emergencia almacenados en los bloques:
 32768, 98304, 163840, 229376, 294912, 819200, 884736
```

```
Escritura de las tablas de i-nodos: completado
Creación del log (32.768 bloques): completado
Escríptura de los superbloques y de la información de compatibilidad del
sistema de ficheros: completado
```

Se verificará el sistema de ficheros automáticamente cada 37 montajes o después de 180 días, según la primera eventualidad. Utilizar **tune2fs -c 0 -i** para sobreescribir el valor.

```
# mount -t ext3 /dev/vg01/data01 /mnt/data01
# df /mnt/data01
Sis. de fich. 1K-blocs Ocupado Disponible Capacidad Montado en
/dev/mapper/vg01-data01
6192704    143488    5734644   3%      /mnt/data01
```

## 5. Ampliaciones y reducciones

## a. Los grupos de volúmenes

Hasta aquí todo sigue el guión esperado. La fuerza del LVM reside en su dinamismo. Vayamos ahora a sacar partido de la situación: supongamos que necesita crear un nuevo LV de 6 GB llamado **data02** en el interior del VG **vg01**. A continuación presentamos el estado actual de **vg01**:

```
# vgdisplay vg01
--- Volume group ---
VG Name           vg01
System ID
Format            lvm2
Metadata Areas    1
Metadata Sequence No 2
VG Access         read/write
VG Status          resizable
MAX LV             0
Cur LV              1
Open LV             1
Max PV              0
Cur PV              1
Act PV              1
VG Size            10,00 GB
PE Size             4,00 MB
```

No hay espacio suficiente, sólo 4 GB (1025 PE). Hay que añadir un nuevo volumen físico en el interior de este VG. Esto se hace con el comando **vgextend**, que funciona de la misma manera que **vgcreate**: indique el nombre del VG seguido del PV o de los PV que quiere añadir.

A continuación se muestra el nuevo estado de **vg01**. Observe que el VG contiene ahora dos PV, y que juntos suman un tamaño de 14 GB (3584 PE).

Sólo queda crear el LV **data02** de 6 GB:

Cuando crea un LV, el LVM busca optimizar el uso de las PE, de manera que sean lo más contiguas posible, y si se puede, en un mismo PV. Esto se ve con el comando **lvdisplay** y el parámetro **-m**, en las líneas **Segments** y la lista de los segmentos propuestos.

Los comandos siguientes crean el sistema de ficheros en el volumen lógico y lo montan:

```
Total PE           2561
Alloc PE / Size   1536 / 6,00 GB
Free PE / Size    1025 / 4,00 GB
VG UUID           dZt8KP-xwol-5mb3-NaVW-Wsui-3sQy-p8kvpG
```

```
# pvcreate /dev/sdb3
Physical volume "/dev/sdb3" successfully created
# vgextend vg01 /dev/sdb3
Volume group "vg01" successfully extended
```

```
# vgdisplay vg01
--- Volume group ---
VG Name           vg01
System ID
Format            lvm2
Metadata Areas    2
Metadata Sequence No 3
VG Access         read/write
VG Status          resizable
MAX LV             0
Cur LV              1
Open LV             1
Max PV              0
Cur PV              2
Act PV              2
VG Size            20,00 GB
PE Size             4,00 MB
Total PE           5120
Alloc PE / Size   1536 / 6,00 GB
Free PE / Size    3584 / 14,00 GB
VG UUID           dZt8KP-xwol-5mb3-NaVW-Wsui-3sQy-p8kvpG
```

```
# lvcreate -n data02 -L 6g vg01
Logical volume "data02" created
```

```
# lvdisplay -m /dev/vg01/data02
File descriptor 3 left open
File descriptor 4 left open
--- Logical volume ---
LV Name           /dev/vg01/data02
VG Name           vg01
LV UUID           QozsEl-tA70-cj2c-RZeD-HfX0-dTm8-0wlYpj
LV Write Access   read/write
LV Status          available
# open             0
LV Size            6,00 GB
Current LE         1536
Segments           1
Allocation         inherit
Read ahead sectors auto
- currently set to 256
Block device       253:1

--- Segments ---
Logical extent 0 to 1535:
  Type            linear
  Physical volume /dev/sdb3
  Physical extents 0 to 1535
```

```
# mkfs -t ext3 /dev/vg01/data02
# mount -t ext3 /dev/vg01/data02 /mnt/data02
```

## b. Ampliar un volumen lógico

Resulta que el LV **data01** de 6 GB es demasiado pequeño. Debe ser el doble. Hay que añadirle 6 GB, lo que es posible porque quedan 8 GB (2048 PE) en el grupo de volúmenes **vg01**:

```
# vgdisplay vg01|grep Free
Free PE / Size    2048 / 8,00 GB
```

La ampliación de un volumen lógico se hace en este orden:

- Ampliación del LV con el comando **lvextend**.
- Ampliación del sistema de ficheros con **resize2fs** (ext)

- o **btrfsctl** (**btrfs**).

### Ampliación del LV

El comando **lvextend** autoriza los parámetros **-l** (número de extensiones lógicas LE) o **-L** como para **lvcreate**. A continuación precise el nuevo tamaño del LV o, si añade un prefijo **+**, el tamaño adicional deseado. También puede precisar, como último argumento, el nombre del PV en el cual forzar la extensión del LV (también es posible con **lvcreate**). Sólo funcionará si el PV o los PV precisados disponen de bastante PE.

El comando siguiente añade 1536 LE (4x1536=6144 MB, o sea, 6 GB) en **data01**:

```
# lvextend -l +1536 /dev/vg01/data01
Extending logical volume data01 to 12,00 GB
Logical volume data01 successfully resized
```

Mire ahora sobre qué PV están ubicados los datos:

El volumen lógico **data01** ocupa, en efecto, 12 GB, en dos segmentos de PE, que están en los PV **/dev/sdb2** y **/dev/sdb3**. El LVM asignó por lo tanto un espacio en el conjunto de los PV del VG.

```
# lvdisplay -m /dev/vg01/data01
--- Logical volume ---
LV Name          /dev/vg01/data01
VG Name          vg01
LV UUID          6ucPwC-sxMJ-K9P3-MkWR-t28I-NyRM-ZKKTmm
LV Write Access  read/write
LV Status        available
# open           1
LV Size          12,00 GB
Current LE       3072
Segments         2
Allocation       inherit
Read ahead sectors auto
- currently set to 256
Block device    253:0

--- Segments ---
Logical extent 0 to 2560:
  Type            linear
  Physical volume /dev/sdb2
  Physical extents 0 to 2560

Logical extent 2561 to 3071:
  Type            linear
  Physical volume /dev/sdb3
  Physical extents 1536 to 2046
```

### Extensión del sistema de ficheros

Sólo el volumen lógico ha sido ampliado. De momento, el tamaño del sistema de ficheros contenido en **data01** no ha cambiado:

El comando **resize2fs** permite reducir y ampliar un sistema de ficheros. El primer argumento es el sistema de ficheros; el segundo, el tamaño, con un eventual sufijo K (KB), M (MB) o G (GB). Sin sufijo, se indica el número de bloques del sistema de ficheros. Si el tamaño está ausente, el sistema de ficheros se adaptará al tamaño de la partición o del LV.

Se puede utilizar el comando **resize2fs** en caliente, es decir, con el sistema de ficheros montado, para las ampliaciones. En cambio, habrá que desmontar el sistema de ficheros para reducirlo.

Mire el estado del sistema de ficheros: ocupa ahora 12 GB:

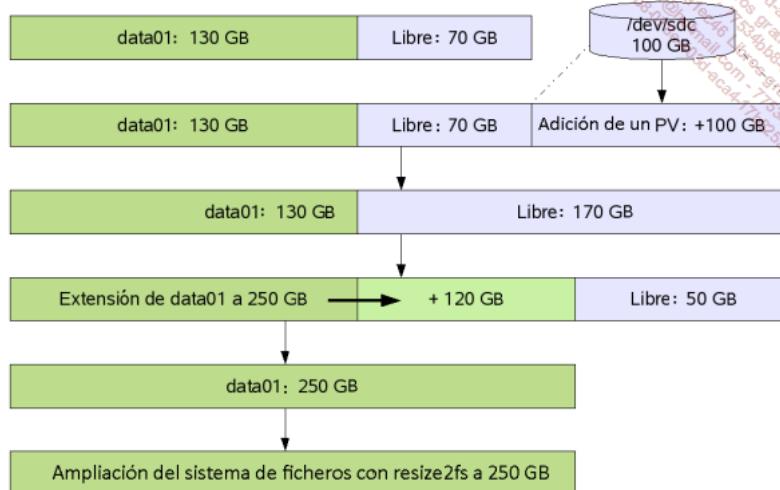
Ahora puede ver la potencia del LVM: adición de nuevos volúmenes físicos y ampliación de volúmenes lógicos en caliente y de manera dinámica. ¿Ya no queda espacio? No importa: basta con añadir un nuevo disco, transformarlo en PV, añadirlo en el VG y volver a dimensionar el LV que no tiene bastante espacio, sin que sea necesario particionar de nuevo, recrear sistemas de ficheros, hacer backups, etc.

```
# df -h /mnt/data01
Sis. de fich. Tam. Oc. Disp. %0c. Montado en
/dev/mapper/vg01-data01 6,0G 141M 5,5G 3% /mnt/data01
```

```
# resize2fs /dev/vg01/data01
resize2fs 1.40.8 (13-Mar-2008)
Filesystem at /dev/vg01/data01 is mounted on /mnt/data01; on-line
resizing required
old_desc_blocks = 1, new_desc_blocks = 1
Performing an on-line resize of /dev/vg01/data01 to 3145728 (4k) blocks.
El sistema de ficheros /dev/vg01/data01 tiene ahora un tamaño
de 3145728 bloques.
```

```
# df -h /mnt/data01
Sis. de fich. Tam. Oc. Disp. %0c. Montado en
/dev/mapper/vg01-data01 12G 144M 12G 2% /mnt/data01
```

Añadir un PV A VG01 y aumentar la capacidad de data01



Un sistema de ficheros btrfs también puede aumentar o reducir su tamaño. Al contrario de un sistema de archivos de tipo ext, la reducción se puede hacer en caliente, sin desmontar el sistema de archivos. Para aumentar sus sistemas de ficheros a la capacidad máxima del LV utilice el comando **btrfsctl**. Observe que se le indica el punto de montaje del sistema de ficheros:

```
# btrfsctl -r max /mnt/data01
```

### c. Reducir un volumen lógico

Para reducir el tamaño de un volumen lógico, tiene que proceder según el orden siguiente:

- Comprobación del sistema de ficheros que hay que reducir con **fsck**, para sistemas de ficheros ext.
- Reducción del sistema de ficheros contenidos en el volumen lógico con **resize2fs** (ext) o **btrfsctl** (btrfs).
- Reducción del volumen lógico con el comando **lvreduce**.

Va a reducir el LV **data01** a 4 GB. Sólo es posible cuando los datos ocupan menos de 4 GB. En un primer momento, compruebe el tamaño actual del sistema de ficheros. Aquí, está casi vacío:

```
# df -h /mnt/data01
Sis. de fich. Tam. Oc. Disp. %0c. Montado en
/dev/mapper/vg01-data01 12G 144M 12G 2% /mnt/data01
```

Sólo se puede reducir el sistema de ficheros si no está montado; desmóntelo:

Compruebe el sistema de ficheros:

Vuelva a dimensionar el sistema de ficheros a 4 GB:

```
# umount /mnt/data01
```

Compruebe el nuevo tamaño del sistema de ficheros. 4096\*1048576 dan, efectivamente, 4 GB.

```
# fsck -f /dev/vg01/data01
fsck 1.40.8 (13-Mar-2008)
e2fsck 1.40.8 (13-Mar-2008)Paso 1: verificación de los i-nodos,
de los bloques y de los tamaños
Paso 2: verificación de la estructura de los directorios
Paso 3: verificación de la conectividad de los directorios
Paso 4: verificación de los contadores de referencia
Paso 5: verificación de la información del sumario del grupo

/dev/vg01/data01: ***** EL SISTEMA DE FICHEROS HA SIDO MODIFICADO *****
/dev/vg01/data01: 11/786432 files (9.1% non-contiguous),
86002/3145728 blocks
```

Para terminar, vuelva a dimensionar el LV a 4 GB. La sintaxis de **lvreduce** es la misma que **lvextend**, excepto que no es posible precisar un PV. Tenga cuidado de no equivocarse aquí: si ha reducido mal el sistema de ficheros, puede destruirlo. Conteste **y** a la pregunta sobre si está seguro.

Monte de nuevo el sistema de ficheros:

```
# resize2fs /dev/vg01/data01 4G
resize2fs 1.40.8 (13-Mar-2008)
Resizing the filesystem on /dev/vg01/data01 to 1048576 (4k) blocks.
El sistema de ficheros /dev/vg01/data01 tiene ahora un tamaño
de 1048576 bloques.
```

```
# dumpe2fs -h /dev/vg01/data01 | grep ^Block
dumpe2fs 1.40.8 (13-Mar-2008)
Block count: 1048576
Block size: 4096
Blocks per group: 32768
```

```
# lvreduce -L 4G /dev/vg01/data01
WARNING: Reducing active logical volume to 4,00 GB
THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce data01? [y/n]: y
Reducing logical volume data01 to 4,00 GB
Logical volume data01 successfully resized
```

```
# mount -t ext3 /dev/vg01/data01 /mnt/data01
# df -h /mnt/data01
Sis. de fich. Tam. Oc. Disp. %0c. Montado en
/dev/mapper/vg01-data01 4,0G 141M 3,7G 4% /mnt/data01
```

### d. Mover el contenido de un volumen físico

En entornos empresariales es corriente mover un PV hacia otro. Puede ser con el objetivo de sustituir un disco que contiene el PV por otro (para ampliar, por ejemplo). En este caso, puede desplazar el contenido de un PV hacia otro, incluso varios PE de un LV hacia otro PV, o también determinados PE precisos. Sin precisar nada como destino, el LVM va a mover todos los PE del PV en los otros PV del grupo de volúmenes. Cuidado: los volúmenes físicos deben ser del mismo grupo de volúmenes.

El comando **pvmove** permite mover los PE de un PV hacia otro. Se trata, para usted, de desplazar el contenido del PV **/dev/sdb3** hacia **/dev/sdb2**. **/dev/sdb3** contiene 1536 PE en uso: todos los LE del LV**data02**.

```
# pvdisplay -m /dev/sdb3
--- Physical volume ---
PV Name          /dev/sdb3
VG Name          vg01
PV Size          10,00 GB / not usable 748,50 KB
Allocatable      yes
PE Size (KByte) 4096
Total PE         2559
Free PE          1023
Allocated PE     1536
PV UUID          Gwk0vR-D0D0-vpA1-zkVb-1Yb2-gcj3-8HbtL6

--- Physical Segments ---
Physical extent 0 to 1535:
 Logical volume   /dev/vg01/data02
 Logical extents  0 to 1535
Physical extent 1536 to 2558:
 FREE
```

Compruebe si el volumen físico **/dev/sdb2** dispone de bastante espacio para recibir el contenido de **/dev/sdb3**. Quedan 1537 PE en este último, por lo tanto es posible.

Mueva el PV **/dev/sdb3** hacia el PV **/dev/sdb2**. Puede utilizar el parámetro **-v** para seguir el progreso. Observe que la operación se efectúa cuando no hay ningún sistema de ficheros desmontado:

Compruebe ahora el estado del grupo de volúmenes:

El segundo PV del **vg01** está totalmente vacío. Por lo tanto, ahora es posible suprimirlo del VG.

```
# pvdisplay /dev/sdb2
--- Physical volume ---
PV Name          /dev/sdb2
VG Name          vg01
PV Size          10,00 GB / not usable 589,00 KB
Allocatable      yes
```

PE Size (KByte)	4096
Total PE	2561
Free PE	1537
Allocated PE	1024
PV UUID	KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra

```
# pvmove -v /dev/sdb3 /dev/sdb2
  Wiping cache of LVM-capable devices
  Finding volume group "vg01"
  Found volume group "vg01"
  Found volume group "vg01"
  Checking progress every 15 seconds
/dev/sdb3: Moved: 4,6%
...
/dev/sdb3: Moved: 97,1%
/dev/sdb3: Moved: 100,0%
  Found volume group "vg01"
  Found volume group "vg01"
  Loading vg01-data02 table
  Suspending vg01-data02 (253:1) without device flush
  Suspending vg01-pvmove0 (253:2) without device flush
  Found volume group "vg01"
  Found volume group "vg01"
  Found volume group "vg01"
  Resuming vg01-pvmove0 (253:2)
  Found volume group "vg01"
  Resuming vg01-data02 (253:1)
  Found volume group "vg01"
  Found volume group "vg01"
  Removing temporary pvmove LV
  Writing out final volume group after pvmove
  Creating volume group backup "/etc/lvm/backup/vg01" (seqno 9).
```

#### e. Reducir un grupo de volúmenes

El comando **vgreduce** permite retirar uno o varios PV de un grupo de volúmenes. Para ello, hace falta primero que los PV en cuestión estén vacíos: sus PE deben estar totalmente libres. Es el caso de `/dev/sdb3`, que va a retirar del VG `vg01`:

Controle que el VG ya no contiene este PV:

```
# vgdisplay -v vg01 | grep -A 100 "Physical"
  Using volume group(s) on command line
  Finding volume group "vg01"
  --- Physical volumes ---
PV Name          /dev/sdb2
PV UUID          KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra
PV Status        allocatable
Total PE / Free PE 2561 / 1

PV Name          /dev/sdb3
PV UUID          Gwk0vR-D0D0-vpA1-zkVb-1Yb2-gcj3-8HbTL6
PV Status        allocatable
Total PE / Free PE 2559 / 2559
```

```
# vgreduce vg01 /dev/sdb3
Removed "/dev/sdb3" from volume group "vg01">@
```

```
# vgdisplay -v vg01 | grep -A 100 "Physical"
  Using volume group(s) on command line
  Finding volume group "vg01"
  --- Physical volumes ---
PV Name          /dev/sdb2
PV UUID          KWfJuL-wBmv-ecD1-u1Wt-Ba3d-KK2b-iryDra
PV Status        allocatable
Total PE / Free PE 2561 / 1
```

## 6. Suprimir un grupo de volúmenes

### a. Etapas

Para suprimir un grupo de volúmenes, debe seguir las etapas siguientes:

- Desmontar todos los sistemas de ficheros de los LV asociados.
- Suprimir todos los volúmenes lógicos con **lvremove**.
- Retirar todos los volúmenes físicos del VG con **lvreduce**.
- Destruir el VG con **vgremove**.

Va a destruir el grupo de volúmenes `vg01`.

### b. Suprimir un volumen lógico

Desmonte `data01` y `data02`:

```
# umount /data01
# umount /data02
```

Suprima los volúmenes lógicos con **lvremove**:

```
# lvremove /dev/vg01/data01 /dev/vg01/data02
Do you really want to remove active logical volume "data01"? [y/n]: y
Logical volume "data01" successfully removed
Do you really want to remove active logical volume "data02"? [y/n]: y
Logical volume "data02" successfully removed
```

### c. Retirar todos los volúmenes físicos

Utilice el comando **vgreduce** con el parámetro `-a` para ALL :

Observe que el comando **vgreduce** siempre deja como mínimo un PV en el VG, porque hace falta al menos un PV para constituir un VG.

```
# vgreduce -a vg01  
Can't remove final physical volume "/dev/sdb2" from volume group  
"vg01"
```

#### d. Destruir un grupo de volúmenes

Utilice el comando **vgremove** para destruir un grupo de volúmenes:

```
# vgremove vg01  
Volume group "vg01" successfully removed
```

Compruebe que los ficheros y directorios asociados han desaparecido:

Para terminar, el comando **vgdisplay** ya no devuelve nada:

```
# ls /dev/vg01  
ls: no puede acceder a /dev/vg01: Ningún fichero o carpeta de este tipo
```

```
# vgdisplay
```

#### e. Suprimir un volumen físico

Ahora se pueden destruir los dos volúmenes físicos, puesto que ya no están en uso. Puede destruir la información contenida en el volumen con el comando **pvremove**. Sin embargo, si destruye la partición vía **fdisk** o crea un sistema de ficheros encima, el efecto es el mismo.

## 7. Comandos adicionales

Sólo hemos echado un vistazo rápido a las posibilidades del LVM. Tenga presente que cuenta con otros comandos, como por ejemplo:

- **pvchange**: modifica el estado de un volumen físico, por ejemplo para prohibir la asignación de extensiones físicas en este volumen.
- **pvresize**: vuelve a dimensionar un volumen físico si su partición o disco de origen ha sido ampliado o reducido.
- **pvscan**: busca todos los volúmenes presentes en todos los soportes de almacenamiento del sistema.
- **vgchange**: modifica los atributos de un grupo de volúmenes, para activarlo o desactivarlo por ejemplo, pero también para modificar sus valores máximos de PV y de PE, o para prohibir su ampliación o su reducción.
- **vgscan**: busca todos los grupos de volúmenes en todos los soportes.
- **vgrename**: renombra un grupo de volúmenes.
- **vgmerge**: ensambla dos grupos de volúmenes en uno único.
- **lvresize**: vuelve a dimensionar un volumen lógico. Equivale tanto a **lvextend** como a **lvreduce**.
- **lvchange**: modifica los atributos de un volumen lógico.
- **lvrename**: renombra un volumen lógico.

# Validación de los conocimientos adquiridos: preguntas/respuestas

## 1. Preguntas

Si cree que sus conocimientos sobre este capítulo son suficientes, conteste a las siguientes preguntas:

### **RAID**

- 1** ¿Qué significa la abreviatura RAID?
- 2** Si un RAID 1 está compuesto de tres discos que disponen individualmente de un MTBF de 30 000 horas. ¿Cuál es el MTBF final?
  - A - 10 000 horas
  - B - 30 000 horas
  - C - 90 000 horas
- 3** Un RAID 5 está compuesto de 4 discos de 100 GB. ¿Cuál es la capacidad total de almacenamiento?
- 4** Dos discos están en RAID 0. El segundo disco está estropeado.
  - A - Los datos aún presentes en el primer disco se conservan.
  - B - Se reparten los datos; por lo tanto, está todo perdido.
  - C - Los datos están presentes en el primer disco; se puede reconstruir el segundo.
- 5** Un RAID5 está compuesto de tres discos y otro de emergencia (spare). Dos discos se estropean con poco tiempo de intervalo. ¿Sigue siendo funcional el RAID?
- 6** ¿Qué comando permite manejar un RAID mediante software?
- 7** ¿Es posible crear un RAID 15 (1+5) mediante software?
- 8** ¿Tiene sentido el comando siguiente?

```
mdadm --create /dev/md0 --level=raid0 --raid-devices=2 /dev/sdb1  
/dev/sdc1 -spare-devices=1 /dev/sdd1
```

- 9** Un cat /proc/mdstat indica esto, ¿qué está pasando?

```
md0: active raid1 hda10[2] hda9[0]F hda8[1]
```

- 10** Normalmente, ¿es posible cambiar un disco IDE en caliente?

### **LVM**

- 11** ¿Qué significa LVM?
- 12** ¿Puede un grupo estar compuesto de volúmenes físicos de diferentes tipos?
- 13** ¿Cuál es el tipo de partición necesaria para constituir un volumen físico?
  - A - fd
  - B - 83
  - C - 82
  - D - 8e
- 14** Si hay tres volúmenes físicos de 30, 70 y 100 GB, ¿cuál es el tamaño máximo de un volumen lógico del grupo de volúmenes constituido por estos tres PV?

**15** Un grupo de volúmenes vgLOCAL está constituido por PE con un tamaño de 64 MB. ¿Cuál es el parámetro que debe facilitarse al comando lvcreate para crear un volumen lógico de 12 GB?

**16** Se desea ampliar lv\_L1 hasta 20 GB. Ahora bien, no hay más PE libre en el VG. ¿Cómo añadir un volumen físico /dev/sde de 20 GB?

- A - vgextend -L +20G /dev/sde vgLOCAL
- B - pvextend /dev/sde vgLOCAL
- C - lvextend -l 320 /dev/vgLOCAL/lv\_L1 /dev/sde
- D - vgextend vgLOCAL /dev/sde

**17** ¿Cuáles de estas sintaxis son correctas para aumentar a 20 GB el volumen lógico lv\_L1?

- A - lvextend -L 20G /dev/vgLOCAL/lv\_L1
- B - lvextend -l +128 /dev/vgLOCAL/lv\_L1
- C - lvextend -l 320 /dev/vgLOCAL/lv\_L1
- D - lvextend -L 20G /dev/vgLOCAL/lv\_L1

**18** A pesar de haber pasado a 20 GB de lv\_L1, un df muestra que el tamaño del sistema de ficheros sigue a 12 GB. ¿De qué se ha olvidado?

**19** Para reducir el tamaño de un sistema de ficheros ext3, ¿qué debe hacer?

**20** Debe cambiar un PV y sustituirlo por otro. ¿Qué comando le permite mover todos los PE de un PV hacia otro?

## 2. Resultados

Diríjase a las páginas siguientes para comprobar sus respuestas. Por cada respuesta correcta, sume un punto.

Número de puntos /20

Para este capítulo, su resultado mínimo debe ser de 15 respuestas acertadas (15/20).

Localice los puntos claves que le dieron problemas y retome su estudio en el capítulo antes de pasar al capítulo siguiente:

- RAID.
- LVM.

## 3. Respuestas

### **RAID**

**1** ¿Qué significa la abreviatura RAID?

*Originalmente, Redundant Array of Inexpensive Disks, pero ahora la explicación corriente es Redundant Array of Independent Disks.*

**2** Si un RAID 1 está compuesto de tres discos que disponen individualmente de un MTBF de 30 000 horas. ¿Cuál es el MTBF final?

- A - 10 000 horas
- B - 30 000 horas

- C - 90 000 horas

*A. Es el resultado siguiente:  $MTBFF = 1 / (1/MTBF1 + 1/MTBF2 + 1/MTBF3 + \dots + 1/MTBFn)$ .*

**3** Un RAID 5 está compuesto de 4 discos de 100 GB. ¿Cuál es la capacidad total de almacenamiento?

*300 GB. Se utiliza el equivalente de un disco (repartido en forma de bandas sobre todos los demás) como control de paridad.*

**4** Dos discos están en RAID 0. El segundo disco está estropeado.

- A - Los datos aún presentes en el primer disco se conservan.
- B - Se reparten los datos; por lo tanto, todo está perdido.
- C - Los datos están presentes en el primer disco; se puede reconstruir el segundo.

*B. Los datos están repartidos en los dos discos.*

**5** Un RAID 5 está compuesto por tres discos y otro de emergencia (spare). Dos discos se estropean con poco tiempo de intervalo. ¿Sigue siendo funcional el RAID?

*Sí. Se sustituye el primer disco HS por el spare. Al estropearse el segundo, las bandas de paridad de los dos primeros permiten acceder (deteriorado en prestaciones) al primero.*

**6** ¿Qué comando permite manejar un RAID de software?

*El comando mdadm.*

**7** ¿Es posible crear un RAID 15 (1+5) mediante software?

*Sí, creando primero al menos tres volúmenes RAID1, luego un RAID5 que compagine los tres volúmenes así creados.*

**8** ¿Tiene sentido el comando siguiente?

```
mdadm --create /dev/md0 --level=raid0 --raid-devices=2 /dev/sdb1
/dev/sdc1 -spare-devices=1 /dev/sdd1
```

*No, este comando no tiene sentido: el RAID 0 no tiene redundancia; por lo tanto, el spare disk no sirve para nada.*

**9** Un cat /proc/mdstat indica esto, ¿qué está pasando?

```
md0: active raid1 hda10[2] hda9[0]F hda8[1]
```

*El disco hda8 que compone el RAID1 está estropeado. Debe sustituirse.*

**10** Normalmente, ¿es posible cambiar un disco IDE en caliente?

*No. Rompería tanto el disco como el controlador o la placa base.*

*La documentación siguiente explica cómo escanear de nuevo una cadena SCSI:*

*<http://kbbase.redhat.com/faq/docs/DOC-3942;jsessionid=7B2A4D06E3CE251ABBFBF5D9484B3B43.ab46478d>*

## **LVM**

**11** ¿Qué significa LVM?

*Logical Volume Manager.*

**12** ¿Puede un grupo estar compuesto de volúmenes físicos de diferentes tipos?

Sí, tanto en soporte (SCSI, IDE, SATA, SAN, etc.) como en tamaño. Sin embargo, observe que se debería asegurar una redundancia para cada volumen físico (por ejemplo, RAID software o hardware).

**13** ¿Cuál es el tipo de partición necesaria para constituir un volumen físico?

- A - fd
- B - 83
- C - 82
- D - 8e

D. El tipo 8e corresponde a «Linux LVM».

**14** Si hay tres volúmenes físicos de 30, 70 y 100 GB, ¿cuál es el tamaño máximo de un volumen lógico del grupo de volúmenes constituido de estos tres PV?

Unos 200 GB (algo de espacio está ocupado en los PV por la estructura del propio LVM).

**15** Un grupo de volúmenes vgLOCAL está constituido por PE con un tamaño de 64 MB. ¿Cuál es el parámetro que debe facilitarse al comando lvcreate para crear un volumen lógico de 12 GB?

Un volumen lógico lv\_L1 se compone de extensiones lógicas del mismo tamaño que las extensiones físicas del grupo de volúmenes que lo contiene. 12 GB corresponden a 12.288 MB. Al dividirlo por 64, se obtienen 192 extensiones. Luego el comando es:

```
lvcreate -l 192 -n lv_L1 vgLOCAL
```

**16** Se desea ampliar lv\_L1 hasta 20 GB. Ahora bien, no hay más PE libre en el VG. ¿Cómo añadir un volumen físico /dev/sde de 20 GB?

- A - vgextend -L +20G /dev/sde vgLOCAL
- B - pvextend /dev/sde vgLOCAL
- C - lvextend -l 320 /dev/vgLOCAL/lv\_L1 /dev/sde
- D - vgextend vgLOCAL /dev/sde

D.

**17** ¿Cuáles de estas sintaxis son correctas para aumentar a 20 GB el volumen lógico lv\_L1?

- A - lvextend -L 20G /dev/vgLOCAL/lv\_L1
- B - lvextend -l +128 /dev/vgLOCAL/lv\_L1
- C - lvextend -l 320 /dev/vgLOCAL/lv\_L1
- D - lvextend -L 20G /dev/vgLOCAL/lv\_L1

Son correctas todas ellas.

**18** A pesar de haber pasado a 20 GB de lv\_L1, un df muestra que el tamaño del sistema de ficheros sigue a 12 Gb. ¿De qué se ha olvidado?

Se ha olvidado de extender el sistema de ficheros. Para un sistema ext3, utilice el comando resize2fs. En los últimos núcleos 2.6.x, el comando se ejecuta en caliente sin necesidad de desmontar el sistema de ficheros.

**19** Para reducir el tamaño de un sistema de ficheros ext3, ¿qué debe hacer?

*Primero debe desmontarlo, luego comprobar los posibles errores con fsck y, por fin, ejecutar el comando resize2fs con el tamaño deseado.*

- 20** Debe cambiar un PV y sustituirlo por otro. ¿Qué comando le permite mover todos los PE de un PV hacia otro?

*El comando pvmove.*

# Prácticas

## 1. Gestionar un RAID1

Objetivo: utilizar los comandos fdisk y mdadm para gestionar un RAID1

1. En su disco duro, cree dos particiones de tamaño idéntico. El tamaño no tiene importancia. Deles el tipo fd, como en el ejemplo siguiente. Diríjase al capítulo Los discos y el sistema de ficheros para la utilización de `fdisk`. Si es necesario, utilice el comando **partprobe** para refrescar la tabla de las particiones del núcleo.

```
# fdisk /dev/sdb
Comando (m para la ayuda): p

Disco /dev/sdb: 160.0 GB, 160041885696 bytes
255 heads, 63 sectors/track, 19457 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x000eab03

Periférico Inicio Principio Fin Bloques Id Sistema
/dev/sdb1 2 16846 135307462+ 83 Linux
/dev/sdb2 16847 18152 10490445 fd Linux raid
autodetect
/dev/sdb3 18153 19457 10482412+ fd Linux raid
autodetect

# partprobe /dev/sdb
```

2. Cree el RAID con el comando **mdadm**. Corresponde a la corta sintaxis siguiente:

```
# mdadm -C /dev/md0 -l 0 -n 2 /dev/sdb2 /dev/sdb3
mdadm: array /dev/md/0 started.
```

3. Compruebe el estado del RAID mirando el contenido de `/proc/mdstat`:

```
# cat /proc/mdstat
Personalities: [raid0]
md0: active raid0 sdb3[1] sdb2[0]
      20972672 blocks 64k chunks
```

4. Compare este resultado con el del comando `mdstat` para una visualización detallada. `/proc/mdstat` proporciona una instantánea del estado del RAID visto desde el núcleo. La salida de `mdstat` proporciona, además del estado actual, la información relativa a la creación y el modo de funcionamiento del RAID:

```
slyserver:/home/seb # mdadm --detail /dev/md0
/dev/md0:
      Version: 0.90
      Creation Time: Wed May 13 20:16:49 2009
      Raid Level: raid0
      Array Size: 20972672 (20.00 GiB 21.48 GB)
      Raid Devices: 2
      Total Devices: 2
      Preferred Minor: 0
      Persistence: Superblock is persistent

      Update Time: Wed May 13 20:16:49 2009
      State: clean
```

```

Active Devices: 2
Working Devices: 2
Failed Devices: 0
Spare Devices: 0

Chunk Size: 64K

        UUID: 3cac5942:d2777be8:1f7b0fb3:3386758f (local to host
slyserver)
        Events: 0.1

Number  Major  Minor  RaidDevice State
      0      8      18          0  active sync  /dev/sdb2
      1      8      19          1  active sync  /dev/sdb3
unused devices: <none>

```

5. Simule una avería del segundo disco. Luego mire el estado del RAID. ¿Sigue en buen estado? Sí, porque el RAID1 es un espejo, los datos son idénticos en los dos discos.

```

# mdadm /dev/md0 -f /dev/sdb3
mdadm: set /dev/sdb3 faulty in /dev/md0
# mdadm --detail /dev/md0|grep State
State: clean

```

6. Pare el RAID y destrúyalo:

```

# mdadm --stop /dev/md0
mdadm: stopped /dev/md0

```

Para destruir de manera definitiva el RAID, hay que suprimir la información de los superbloques de los discos que componen el RAID. Es ahí donde se encuentra escrita la información sobre el RAID. Se hace esto con el parámetro `--zero-superblock`:

```
# mdadm --zero-superblock /dev/sdb2 /dev/sdb3
```

## 2. Trabajar con un LVM

Objetivo: Trabajar con PV, VG y LV.

1. Modifique las particiones del TD anterior con el tipo 8e:

/dev/sdb2	16847	18152	10490445	8e	Linux	LVM
/dev/sdb3	18153	19457	10482412+	8e	Linux	LVM

2. Cree dos PV con estas dos particiones:

```

# pvcreate /dev/sdb2
Physical volume "/dev/sdb2" successfully created
# pvcreate /dev/sdb3
Physical volume "/dev/sdb3" successfully created

```

3. Cree un grupo de volúmenes vgLOCAL con el primer PV:

```

slyserver:/home/seb # vgcreate vgLOCAL /dev/sdb2
Volume group "vgLOCAL" successfully created

```

4. Compruebe el estado del VG. En particular, observe el tamaño de un Physical Extend:

```
# vgdisplay vgLOCAL
--- Volume group ---
VG Name          vgLOCAL
System ID
Format           lvm2
Metadata Areas   1
Metadata Sequence No 1
VG Access        read/write
VG Status         resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size          10,00 GB
PE Size        4,00 MB
Total PE         2561
Alloc PE / Size  0 / 0
Free  PE / Size  2561 / 10,00 GB
VG UUID          VYNM1X-D4al-PPZl-oD4t-7l5r-HyPP-SdZuDw
```

5. Cree un volumen lógico lv\_L1 de 8 GB (adáptelo según el tamaño de su VG), utilizando los Logical extends. Aquí, hacen falta 2 048 extensiones. A continuación, dé formato al LV en ext3.

```
# lvcreate -n lv_L1 -l 2048 vgLOCAL
Logical volume "lv_L1" created

# mkfs -t ext3 /dev/vgLOCAL/lv_L1
mke2fs 1.41.1 (01-Sep-2008)
...
Escritura de los superbloques y de la información de contabilidad
del sistema de ficheros: completado
```

6. Amplíe lv\_L1 a 15 GB. Para ello, añada el segundo PV en el grupo de volúmenes, luego amplíe el LV y, por fin, amplíe el sistema de ficheros.

```
# vgextend vgLOCAL /dev/sdb3
Volume group "vgLOCAL" successfully extended
# lvextend -L +3G /dev/vgLOCAL/lv_L1
Extending logical volume lv_L1 to 11,00 GB
Logical volume lv_L1 successfully resized
# resize2fs /dev/vgLOCAL/lv_L1
resize2fs 1.41.1 (01-Sep-2008)
Resizing the filesystem on /dev/vgLOCAL/lv_L1 to 2883584 (4k) blocks.
El sistema de ficheros /dev/vgLOCAL/lv_L1 tiene ahora un tamaño
de 2883584 bloques.
```

7. Reduzca el volumen lógico lv\_L1 a 5 GB. Para ello, reduzca primero el sistema de ficheros a este tamaño, luego el volumen lógico. Tendrá que confirmar esta última operación, ya que puede ser destructiva si se olvida de reducir el tamaño del sistema de ficheros.

```
# resize2fs /dev/vgLOCAL/lv_L1 5G
resize2fs 1.41.1 (01-Sep-2008)
Resizing the filesystem on /dev/vgLOCAL/lv_L1 to 1310720 (4k) blocks.
El sistema de ficheros /dev/vgLOCAL/lv_L1 tiene ahora un tamaño de
1310720 bloques.

# lvreduce -L 5G /dev/vgLOCAL/lv_L1
```

```
WARNING: Reducing active logical volume to 5,00 GB  
THIS MAY DESTROY YOUR DATA (filesystem etc.)  
Do you really want to reduce lv_L1? [y/n]: y  
Reducing logical volume lv_L1 to 5,00 GB  
Logical volume lv_L1 successfully resized
```

8. Para terminar, destruya totalmente el grupo de volúmenes. Empiece por destruir el volumen lógico lv\_L1 y todos los volúmenes lógicos que puedan estar presentes. Luego, retire todos los volúmenes físicos del grupo de volúmenes, salvo uno. Para terminar, destruya el grupo de volúmenes.

```
# lvremove /dev/vgLOCAL/lv_L1  
Do you really want to remove active logical volume "lv_L1"? [y/n]: y  
Logical volume "lv_L1" successfully removed  
# vgreduce vgLOCAL /dev/sdb3  
Removed "/dev/sdb3" from volume group "vgLOCAL"  
# vgremove vgLOCAL  
Volume group "vgLOCAL" successfully removed
```

## Modelo SQL (para el capítulo El shell y los comandos GNU)

Este modelo no es el más óptimo. De hecho, se deberían separar los pedidos en dos tablas: una tabla de pedidos correspondiente a un número único de pedido y toda la información relacionada con ella EXCEPTO los productos (id\_pedido, id\_cliente, tipo de entrega, posible dirección de entrega si es diferente de la del cliente, fecha del pedido, estado, empleado que lo procesó, fecha de envío, tipo de envío, etc.) y una tabla detalle\_pedido relacionando el número de pedido y el número de producto, la cantidad, un posible descuento, etc.

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

-- Estructura de la tabla `t_pedido`

CREATE TABLE IF NOT EXISTS `t_pedido` (
  `id_pedido` int(11) NOT NULL auto_increment,
  `id` int(11) NOT NULL,
  `id_producto` int(11) NOT NULL,
  `qte` int(11) NOT NULL,
  PRIMARY KEY (`id_pedido`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

-- Contenido de la tabla `t_pedido`

INSERT INTO `t_pedido` (`id_pedido`, `id`, `id_producto`, `ctd`) VALUES
(1, 1, 3, 1),
(2, 1, 1, 2),
(3, 2, 2, 5);

-- Estructura de la tabla `t_proveedores`

CREATE TABLE IF NOT EXISTS `t_proveedores` (
  `id_proveedor` int(11) NOT NULL auto_increment,
  `apellido` varchar(255) NOT NULL,
  `dirección` varchar(255) NOT NULL,
  `cp` varchar(5) NOT NULL,
  `ciudad` varchar(100) NOT NULL,
  `tel` varchar(10) NOT NULL,
  PRIMARY KEY (`id_proveedor`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

-- Contenido de la tabla `t_proveedores`

INSERT INTO `t_proveedores` (`id_proveedor`, `apellido`, `dirección`, `cp`,
`ciudad`, `tel`) VALUES
(1, 'Espacio Multimedia', 'Calle de los pájaros 48', '77120', 'Madrilona',
'160606161'),
(2, 'las verduras asociadas', 'Avenida de los horticultores 14', '28140',
'Lechugas de Madrid', '321414141');

-- Estructura de la tabla `t_productos`

CREATE TABLE IF NOT EXISTS `t_productos` (
  `id_producto` int(11) NOT NULL auto_increment,
  `id_proveedor` int(11) NOT NULL,
  `nombre` varchar(150) NOT NULL,
  `precio` float NOT NULL,
  `iva` float NOT NULL,
  `ctd` int(11) NOT NULL,
  PRIMARY KEY (`id_producto`),
  KEY `id_proveedor`(`id_proveedor`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

-- Contenido de la tabla `t_productos`
```

```
INSERT INTO `t_productos` (`id_producto`, `id_proveedor`, `nombre`, `precio`,  
`iva`, `ctd`) VALUES  
(1, 1, 'Llave USB 1 Go', 10.71, 1.196, 20),  
(2, 1, 'piña primera calidad', 16.002, 1.196, 30),  
(3, 2, 'Rioja Bueno 2009', 14.9835, 1.196, 25);  
  
-- Estructura de la tabla `t_usuarios`  
  
CREATE TABLE IF NOT EXISTS `t_usuarios` (  
  `id` int(11) NOT NULL auto_increment,  
  `apellido` varchar(100) NOT NULL,  
  `nombre` varchar(100) NOT NULL,  
  `ciudad` varchar(100) NOT NULL,  
  `id_alias` int(11) default NULL,  
  PRIMARY KEY (`id`),  
  KEY `id_alias` (`id_alias`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;  
  
-- Contenido de la tabla `t_usuarios`  
  
INSERT INTO `t_usuarios` (`id`, `apellido`, `nombre`, `ciudad`, `id_alias`)  
VALUES  
(1, 'García', 'Santiago', 'Madrid', NULL),  
(2, 'Aguilera', 'Francisco', 'Alcorcón', 1);
```

# Objetivos del examen LPI 101

Se pondera cada objetivo con un número (valor indicado entre paréntesis) del 1 al 10, e indica la importancia relativa de cada objetivo. Cuanto mayor sea el número atribuido a un objetivo, más preguntas sobre ese objetivo aparecerán en el examen.

	Capítulo	Prácticas
<b>101 Hardware y arquitectura</b>		
101.1 Ajustes fundamentales de la BIOS (2)	Inicio de Linux, servicios, núcleo y periféricos - Proceso de inicio - La BIOS	
101.2 Poner en marcha el sistema (3)	Inicio de Linux, servicios, núcleo y periféricos - Proceso de inicio	Inicio de Linux, servicios, núcleo y periféricos - GRUB y el proceso de boot
101.3 Cambiar los niveles, parar o reiniciar (3)	Inicio de Linux, servicios, núcleo y periféricos - init	Inicio de Linux, servicios, núcleo y periféricos - init y runlevel
<b>102 Instalación de Linux y gestión de los paquetes</b>		
102.1 Preparar el disco duro (2)	Instalación de Linux y de los programas : <ul style="list-style-type: none"><li>• Instalar una DEBIAN<ul style="list-style-type: none"><li>- Particionar los discos</li></ul></li><li>• Instalación de openSUSE - Particionamiento</li></ul> Los discos y el sistema de ficheros - Representación de los discos	Instalación de Linux y de los programas - Esquema de particionamiento  Los discos y el sistema de ficheros - Los discos y particiones
102.2 Instalar un gestor de carga (2)	Instalación de Linux y de los programas - Instalar una DEBIAN - Fin de instalación y reinicio  Inicio de Linux, servicios, núcleo y periféricos - Proceso de inicio - GRUB	Inicio de Linux, servicios, núcleo y periféricos - GRUB y el proceso de boot
102.3 Gestionar las librerías compartidas (1)	Instalación de Linux y de los programas - Gestionar las librerías compartidas	Instalación de Linux y de los programas - Librerías compartidas
102.4 El Gestor de paquetes Debian y APT (3)	Instalación de Linux y de los programas: <ul style="list-style-type: none"><li>• Debian Package</li></ul>	Instalación de Linux y de los programas - Gestion de DPKG y APT

	<ul style="list-style-type: none"> <li>• Gestor APT</li> </ul>	
102.5 El gestor de paquetes RPM y YUM (3)	<p>Instalación de Linux y de los programas:</p> <ul style="list-style-type: none"> <li>• Red Hat Package Manager</li> <li>• YUM</li> </ul>	Instalación de Linux y de los programas - Gestión de los RPM

### **103 Los comandos GNU y Unix**

103.1 Usar la línea de comandos (4)	El shell y los comandos GNU - El shell bash	
103.2 Tratar los flujos texto con los filtros (3)	El shell y los comandos GNU - Filtros y herramientas	El shell y los comandos GNU - Los filtros y herramientas
103.3 Gestión básica de los ficheros (3)	El shell y los comandos GNU - La gestión de los ficheros	El shell y los comandos GNU - Gestión de los ficheros
103.4 Canales, tuberías y redirecciónamientos (4)	El shell y los comandos GNU - Redirecciones	El shell y los comandos GNU - Las redirecciones
103.5 Crear, visualizar y matar los procesos (4)	El shell y los comandos GNU - Los procesos	El shell y los comandos GNU - Los procesos
103.6 Modificar la prioridad de los procesos (2)	El shell y los comandos GNU - Los procesos - nice y renice	El shell y los comandos GNU - Los procesos
103.7 Buscar texto, expresiones regulares (2)	El shell y los comandos GNU - Buscar ficheros	El shell y los comandos GNU - Los filtros y herramientas
103.8 Los fundamentos del editor vi (3)	El shell y los comandos GNU - El editor vi	

### **104 Periféricos de almacenamiento, sistema de ficheros, estructura en árbol**

104.1 Crear particiones y sistemas de ficheros (2)	<p>Los discos y el sistema de ficheros:</p> <ul style="list-style-type: none"> <li>• Operaciones de bajo nivel</li> <li>• Elegir un sistema de ficheros</li> <li>• Particionamiento</li> <li>• Manejar los sistemas de ficheros</li> <li>• La swap</li> </ul>	<p>Los discos y el sistema de ficheros:</p> <ul style="list-style-type: none"> <li>• Los discos y particiones</li> <li>• Creación de un sistema de ficheros</li> </ul>
----------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------

104.2 Mantener la integridad de los sistemas de ficheros (2)	Los discos y el sistema de ficheros - Controlar el sistema de ficheros	Los discos y el sistema de ficheros - Estadísticas y mantenimiento del sistema de ficheros
104.3 Montar y desmontar sistemas de ficheros (3)	Los discos y el sistema de ficheros - Acceder a los sistemas de ficheros	Los discos y el sistema de ficheros - Acceso y montaje del sistema de ficheros
104.4 Aplicar cuotas (1)	Los discos y el sistema de ficheros - Las cuotas de disco	Los discos y el sistema de ficheros - Cuotas
104.5 Modificar los permisos sobre ficheros (3)	Los discos y el sistema de ficheros - Los permisos de acceso: <ul style="list-style-type: none"> <li>• Los permisos básicos</li> <li>• Modificación de los permisos</li> <li>• Máscara de permisos</li> <li>• Cambiar de propietario y de grupo</li> <li>• Permisos de acceso especiales</li> </ul>	Los discos y el sistema de ficheros - Los permisos
104.6 Crear y modificar los vínculos (2)	Los discos y el sistema de ficheros - Manejar los sistemas de ficheros - Definición básica - Hard link El shell y los comandos GNU - La gestión de los ficheros - Los comandos básicos - Gestionar los ficheros y los directorios	
104.7 Buscar ficheros (2)	El shell y los comandos GNU - Buscar ficheros	El shell y los comandos GNU - Buscar ficheros

# Objetivos del examen LPI 102

<b>105 Shell, scripts, programación y compilación</b>		
105.1 Modificar el entorno del shell (4)	Las tareas administrativas - Administración de los usuarios - El entorno de usuario  El shell y los comandos GNU - Configuración del bash	
105.2 Escribir y modificar scripts (4)	El shell y los comandos GNU - Programación shell	El shell y los comandos GNU: <ul style="list-style-type: none"><li>• Programación Shell Nivel 1</li><li>• Función Shell</li></ul>
105.3 Gestión de datos con SQL	El shell y los comandos GNU - SQL	
<b>106 Entornos gráficos y de escritorio</b>		
106.1 Instalar y configurar X11 (2)	X Window - Xorg	X Window - Configuración simple de Xorg
106.2 Configurar un visualizador (2)	X Window - El Display Manager	X Window - Configuración simple del Display Manager
106.3 Accesibilidad (1)	X Window - Accesibilidad	
<b>107 Tareas administrativas</b>		
107.1 Gestionar los grupos, los usuarios y los directorios personales (5)	Las tareas administrativas - Administración de los usuarios	Las tareas administrativas - Gestión de los usuarios
107.2 Automatizar las tareas de administración (4)	Las tareas administrativas - Automatización	Las tareas administrativas - Automatización de tareas
107.3 Regionalización e internacionalización (3)	Las tareas administrativas - Los parámetros regionales	
<b>108 Servicios esenciales del sistema</b>		
108.1 Mantener el sistema en hora (3)	Las tareas administrativas - El reloj	
108.2 Configurar syslog (2)	Las tareas administrativas - Los registros (logs) del	Las tareas administrativas - Las trazas del sistema

	sistema	
108.3 Configuración básica del MTA (3)	La red - Correo electrónico	La red - Servicios de red
108.4 Gestionar las impresoras y colas de impresión (2)	Las tareas administrativas - La impresión	Las tareas administrativas - La impresión
<b>109 Las bases de la red</b>		
109.1 Las bases de TCP/IP (4)	<p>La red - TCP/IP:</p> <ul style="list-style-type: none"> <li>• Fundamentos</li> <li>• Direccionamiento</li> </ul>	La red - Configuración TCP/IP de Linux
109.2 Configuración TCP/IP de Linux (4)	<p>La red - TCP/IP:</p> <ul style="list-style-type: none"> <li>• Configuración</li> <li>• Herramientas de red</li> <li>• Ficheros generales</li> </ul>	<p>La red:</p> <ul style="list-style-type: none"> <li>• Configuración TCP/IP de Linux</li> <li>• Algunos comandos de red</li> </ul>
109.3 Resolver los problemas de red comunes	La red - TCP/IP	
109.4 Gestión de los nombres de dominios DNS básica (4)	La red - Servidor DNS	La red - El resolver
<b>110 Seguridad</b>		
110.1 Tareas de administración de seguridad (3)	La seguridad - Bases de seguridad	La seguridad - Control de los ficheros
110.2 Seguridad de la máquina local (3)	<p>La seguridad:</p> <ul style="list-style-type: none"> <li>• Bases de seguridad</li> <li>• Seguridad de servicios y de red</li> </ul>	<p>La seguridad:</p> <ul style="list-style-type: none"> <li>• Control de los ficheros</li> <li>• Seguridad general del sistema</li> <li>• Seguridad de red</li> </ul>
110.3 Proporcionar mayor seguridad a los datos mediante encriptación	<p>La red - OpenSSH</p> <p>La seguridad - Seguridad de servicios y de red - GPG</p>	La seguridad - Seguridad de los usuarios